

Automated Recovery of Three-Dimensional Models of Plant Shoots from Multiple Color Images¹[C][W][OPEN]

Michael P. Pound, Andrew P. French, Erik H. Murchie, and Tony P. Pridmore*

Centre for Plant Integrative Biology, School of Biosciences, University of Nottingham, Sutton Bonington LE12 5RD, United Kingdom (M.P.P., A.P.F., E.H.M., T.P.P.); and School of Computer Science, University of Nottingham, Jubilee Campus, Nottingham NG8 1BB, United Kingdom (A.P.F., T.P.P.)

Increased adoption of the systems approach to biological research has focused attention on the use of quantitative models of biological objects. This includes a need for realistic three-dimensional (3D) representations of plant shoots for quantification and modeling. Previous limitations in single-view or multiple-view stereo algorithms have led to a reliance on volumetric methods or expensive hardware to record plant structure. We present a fully automatic approach to image-based 3D plant reconstruction that can be achieved using a single low-cost camera. The reconstructed plants are represented as a series of small planar sections that together model the more complex architecture of the leaf surfaces. The boundary of each leaf patch is refined using the level-set method, optimizing the model based on image information, curvature constraints, and the position of neighboring surfaces. The reconstruction process makes few assumptions about the nature of the plant material being reconstructed and, as such, is applicable to a wide variety of plant species and topologies and can be extended to canopy-scale imaging. We demonstrate the effectiveness of our approach on data sets of wheat (*Triticum aestivum*) and rice (*Oryza sativa*) plants as well as a unique virtual data set that allows us to compute quantitative measures of reconstruction accuracy. The output is a 3D mesh structure that is suitable for modeling applications in a format that can be imported in the majority of 3D graphics and software packages.

In recent years, there has been a surge of interest in the construction of geometrically accurate models of plants. Increased adoption of the systems approach to biological research has focused attention on the use of quantitative models of biological objects and processes to both make and test hypotheses. Applications of the systems approach are diverse and include the study of canopy photosynthesis (Watanabe et al., 2005; Song et al., 2013). The work reported here was motivated by a need for realistic three-dimensional (3D) representations of plant shoots for photosynthesis modeling. Canopy structure or architecture is an important codeterminant of the maximum productivity of crops and theoretically can influence canopy photosynthesis efficiency (Zhu et al., 2004; Long et al., 2006; Reynolds et al., 2012). For example, many high-yielding cereal varieties tend to have upright leaves, which raises the optimal leaf area index and reduces the proportion of leaves in a light-saturated condition (Murchie and Reynolds, 2012). However, there is

still considerable variation in plant canopy architecture, and it is currently unclear whether many existing architectures (which have been influenced by breeding for specific traits) offer optimal arrangements for photosynthesis (Long et al., 2006; Murchie et al., 2009). Therefore, there is a need for rapid, automated, user-accessible techniques for accurately measuring the 3D architecture of complex crop canopies.

Existing plant modeling approaches can be broadly classified as either rule based or image based (Quan et al., 2006). Rule-based approaches generate model plants based on rules or grammars with specified structure. These rules, and hence the form and parameters of the models produced, are often derived from measurements of real plants (Watanabe et al., 2005; Alarcon and Sassenrath, 2011). The resulting virtual plants can model different phenotypes, plant response to various growing conditions and stresses, and when based on real-world data will be reasonably accurate. However, the data acquisition process is often extremely time consuming and is usually tailored to a particular species. In many cases, only a small set of varieties can be described, due to the manual measurements required to parameterize the model. It is also difficult to incorporate additional data into a rule-based model after it has been created. For example, modeling the effect of environmental factors on plant development will require rules to be modified and additional measures to be taken, again a time-consuming process. Critically, while models created in this way may capture the important characteristics of a given species, they do not necessarily describe any specific, individual plant. This is important when considering the variation in plant structure that can occur in response to environment

¹ This work was supported by the Biotechnology and Biological Sciences Research Council (funding to the Centre for Plant Integrative Biology to E.H.M. and T.P.P.).

* Address correspondence to tony.pridmore@nottingham.ac.uk.

The author responsible for distribution of materials integral to the findings presented in this article in accordance with the policy described in the Instructions for Authors (www.plantphysiol.org) is: Tony P. Pridmore (tony.pridmore@nottingham.ac.uk).

[C] Some figures in this article are displayed in color online but in black and white in the print edition.

[W] The online version of this article contains Web-only data.

[OPEN] Articles can be viewed online without a subscription.

www.plantphysiol.org/cgi/doi/10.1104/pp.114.248971

and features that are associated with communities of plants, including crop canopies. This can be seen in the relationship between planting density and tillering rate, for example (Zhong et al., 2002).

Image-based approaches attempt to directly model a given object by extracting the necessary information from one or more images of that object. The approach relies on techniques developed in the wider field of computer vision and is computationally challenging. The continued increase in available computing power, however, means that this can be done efficiently, and in some cases fully automatically. Image-based approaches to plant modeling are particularly attractive, as, in addition to supporting systems biology, they provide a route to plant phenotyping (Houle et al., 2010; White et al., 2012). 3D geometrical models contain the information needed to compute summary plant traits, such as total leaf area, mean leaf angle, etc. These underpin both plant breeding programs and attempts to understand the relationship between genotype, phenotype, and environment, regardless of the scientific approach taken.

Plants, however, provide a particularly challenging image-based modeling target, including large amounts of self-occlusion (leaves obscuring one another) and many smaller surfaces that appear similar. Depending on the plant species, leaves can lack the texture necessary to perform robust feature matching, either to separate leaves from one another or to locate specific leaves across multiple views. To overcome this challenge, where image-based modeling approaches are successful, they have often involved user interaction to guide the process (Quan et al., 2006).

It is possible to further categorize image-based approaches into those that are top down, beginning with an existing, generic plant model that is fitted to the received image data, and those that are bottom up, creating a description of the plant by examining the contents of the images. Top-down approaches attempt to simplify the model construction problem by instead solving a model refinement problem. An existing model is adjusted to fit the image data, so that the new plant representation is consistent with what is observed. Quan et al. (2006) take this approach, first obtaining an ideal leaf model from a single leaf and then fitting it to all other (user-segmented) leaves in the scene. By adapting an existing model, topological inconsistency (such as the self-intersection of leaf surfaces) is avoided, but this comes at the expense of generality. This approach can be extended in principle to other plant species, but where the geometry of a plant or leaf differs greatly from the expected model, the suitability of this approach is unclear. This is also significant where a plant model has been generated with highly specific real-world data, such as the work of Watanabe et al. (2005).

Bottom-up methods begin with one or more images and reconstruct a plant model based only on the observed pixel data. Two broad approaches exist, both requiring a set of images captured from different, but known, viewpoints. Silhouette-based methods (Clark et al., 2011; Kumar et al., 2012) segment each image to

identify the boundary of the object of interest. Each silhouette is effectively projected into the viewed environment, identifying a region of 3D space that could be occupied by the plant. These regions are combined to determine the maximum possible object size that is consistent with the images presented to the algorithm. In many cases, where the number of input images is high, the resulting model will be a good approximation of the true plant structure. However, as the scene becomes increasingly complex, for example with the addition of more leaves in an older plant, the discrepancy between the true object and the model will increase. This problem becomes more pronounced when extending these techniques to very complex scenes, such as plant canopies, where its effectiveness is limited.

Correspondence-based methods identify features of interest (i.e. recognizable patches of pixels in the image), independently in each of a set of images, and then match those features between views. If the image features associated with a particular plant feature (e.g. the tip of a leaf) can be identified in multiple images taken from different viewpoints, knowledge of the cameras' positions and orientations allow its 3D location to be computed. The result is a point cloud representation, a set of (x, y, z) coordinates approximating the plant's surface (Quan et al., 2006; Omasa et al., 2007; Alenya et al., 2011). Point cloud data can be obtained directly from special-purpose hardware devices such as light detection and ranging laser (LIDAR; Omasa et al., 2007) and time-of-flight laser (Alenya et al., 2011) scanners. These systems use a large number of distance samples to measure the depth of the scene relative to the capture location. The equipment involved, however, can be expensive and often places restrictions on the environments in which it can operate.

Image-based modeling algorithms are widely applicable and require only easily accessible and affordable cameras. Their generality, however, can become a hindrance, as the challenging nature of plant topology may require additional assumptions to be made as the reconstruction proceeds. The representations they produce may also be unsuitable for direct use in some situations. The volumetric data structures produced by silhouette-based methods, for example, are static: the size and positions of the voxels are defined early in the process and are difficult to change. While measurements of height and volume are easily made from volumetric descriptions, estimating motion (e.g. of leaves moving in the breeze) is extremely difficult. Volume-based descriptions contain an implicit surface around that volume that can be measured; however, these surfaces are less suitable for ray tracing and modeling, as each leaf is represented as numerous surfaces (inside, outside, and sides of the leaf volume). Similarly, point clouds can be used to calculate the density and distribution of plant material but cannot immediately be used to host models of photosynthetic activity; for this, a surface-based representation is required.

This article describes a fully automatic, bottom-up approach to image-based 3D plant reconstruction that is applicable to a wide variety of plant species and topologies. The method is accurate, providing a true representation of

the original plant, and produces data in a form that can support both trait measurements and modeling techniques such as forward ray tracing (Song et al., 2013).

The proposed approach requires a point cloud and a set of color images. We obtain these using existing correspondence-based techniques (Furukawa and Ponce, 2010; Wu, 2011), with a focus on inexpensive equipment; however, hardware-based approaches that generate point clouds in three dimensions, such as LIDAR scanners, also could be used. The point cloud is first described by fitting a set of planar patches (flat surfaces), each representing a small section of plant material, usually a segment of leaf. Where the quality of the input point cloud is high, the initial surface estimate will provide a good model of the plant. Image noise and the complexity of the plant, however, will typically lead to missing areas of leaf material and poorly defined leaf boundaries. Therefore, we extend existing approaches by refining the initial surface estimate into a more accurate plant model. Initial surface patches are resized and reshaped based on image information and information obtained from neighboring surfaces. The resulting surfaces are then subdivided into triangular sections suitable for output to produce a smooth and geometrically accurate model of the plant.

The reconstruction process makes few assumptions about the nature of the plant material being reconstructed; by representing each leaf as a series of small, flat sections (like pieces in a jigsaw puzzle), the complete leaf surface itself can take any reasonable shape. The output is a 3D triangular mesh structure that is suitable for ray tracing or other modeling applications in a format that can be imported into many 3D graphics and modeling packages. The generality of our technique allows it to be scaled to scenes involving multiple plants and even plant canopies: the software makes no assumptions about the number of plants in the image. However, the focus of this article is on the accurate reconstruction of single plants of varying species. In the discussion below, we will outline our intentions for extending this method to field-scale phenotyping.

A software tool utilizing the techniques described in this article has been released as an open-source project under Berkeley Software Distribution license.

RESULTS

In this section, we present results obtained when applying our reconstruction approach to multiple views of single plants. Verification of our approach is achieved using a unique artificial data set in which an *in silico* model rice plant is rendered from multiple viewpoints to generate artificial color images that are then treated in the same way as a real-world image set. This approach allows the reconstructed plant to be directly compared with the artificial target object, a difficult problem if no such ground truth were to exist.

We have tested our reconstruction methods on data sets obtained from rice (*Oryza sativa*) and wheat (*Triticum aestivum*) plants. The number and nature of the images were left to the user to decide given the subject in question, although we recommend more than 30 images surrounding the subject for a single plant, including a variety of angles from side on to above. No special consideration was given to the environment in which the plants were imaged: the rice data set was captured in an indoor environment and the wheat data set in a greenhouse. These environments provide complex backgrounds, which raise additional challenges, but the plants can still be reconstructed using our methods. It is likely that a permanent installation with a more strict protocol for image capture would result in more consistent point cloud reconstruction between data sets; readers are encouraged to explore this option if using our methods over extended periods. An overview of each data set is given in Table I.

The computational performance of our approach and the image capture that proceeds it varies depending on the number of images and the parameters used. For a recommended data set of 30 or more images using default parameters, both the image capture and reconstruction can be expected to take on the order of a few minutes.

Reconstruction of Example Rice and Wheat Data Sets

Figure 1 shows the result of applying our reconstruction approach to the rice and wheat data sets. The reconstructions are colored based on the normal orientation of each surface; the models have not been

Table I. Descriptions of the single plant data sets used to test our reconstruction approach

The rice and wheat data sets were captured manually using a single camera. The virtual rice data set was generated using 3D modeling software based on a template rice plant created manually. In all cases, the input into our reconstruction software was the calibrated camera coordinates generated by VisualSFM, the set of input images, and an initial point cloud generated by PMVS.

| Data Set Name | Image Count | Description |
|---------------|-------------|---|
| Rice | 36 | Images of a rice plant (variety IR64) at leaf 10 stage held in a cylindrical flask; images were captured from a variety of angles surrounding the plant and from above |
| Wheat | 64 | Images of a wheat plant at growth stage 37 held statically in a growth pot; images were taken in concentric movements of increasing height around the plant, at a higher frequency than that of the rice data set |
| Virtual rice | 40 | A model plant was manually created based on the rice data set; this model was used to generate 40 artificial images from a variety of viewpoints surrounding and above the plant; unlike real-world examples, the ground truth of this model is known and can be compared with the output of the reconstruction process |

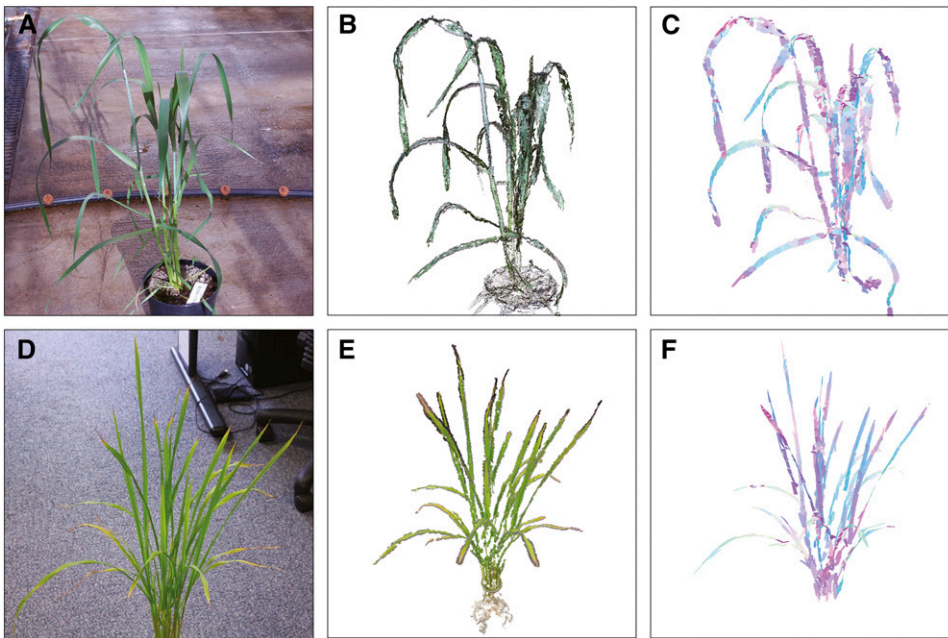


Figure 1. Output of our reconstruction approach. A, An image of a wheat plant from the wheat data set, a multiview data set captured in a glasshouse. B, A point cloud representation of the wheat data set output by the PMVS software. C, A reconstructed surface mesh of the complete wheat plant produced by our algorithm. D, An image of a rice plant from the rice data set, a multiview data set captured in an office environment. E, A point cloud representation of the rice data set output by the PMVS software. F, A reconstructed surface mesh of the complete rice plant produced by our algorithm. Both completed meshes have been colored based on surface orientation for clarity.

textured to avoid concealing imperfections in the output mesh. Quantitative evaluation of the effectiveness of any 3D shoot reconstruction is challenging due to a lack of ground truth models for comparison. Here, we offer a qualitative evaluation of the benefits and shortcomings of our approach followed by a quantitative evaluation using the virtual rice data set.

The initial surface estimate, obtained by calculating an α -shape over each patch, will naturally reproduce any flaws present in the patch-based multiple-view stereo (PMVS) point cloud (Fig. 2, A and C). Most notable is the lack of point information in areas of poor texture and noise perpendicular to the leaf surface, where depth has not been adequately resolved. These issues can be caused by the heavy self-occlusion observed in larger plants but

are often caused in even simple data sets by a lack of image features in the center of leaves.

Depth noise is significantly reduced by the use of best-fit planes in small patches, as all points are projected onto a single flat surface. However, the boundary of each surface is a function of the parameters used to create the α -shape and the quality of the underlying data. As such, we can expect the α -shape boundaries to be a poor representation of the true leaf shape. With this in mind, we would characterize a successful reconstruction as one that improves upon the initial surface estimate through the optimization of the surface boundaries.

Consider Figure 2, A and C, depicting initial surface estimates from the rice and wheat plants in Figure 1. In Figure 2A, the initial surface contains a great deal of

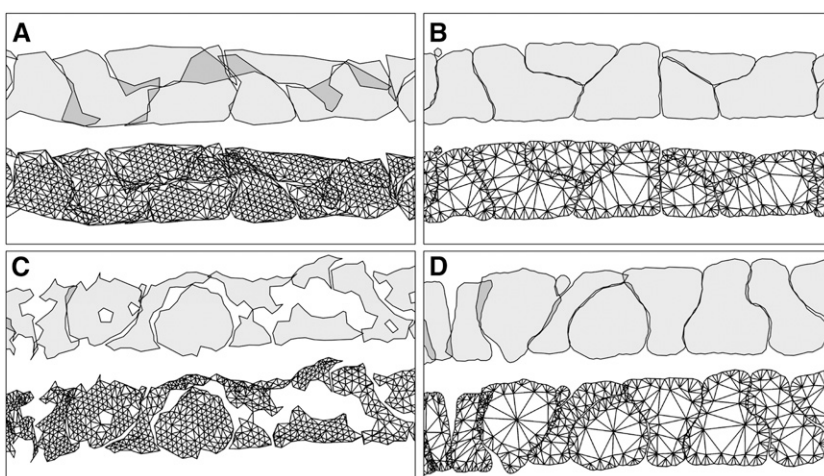
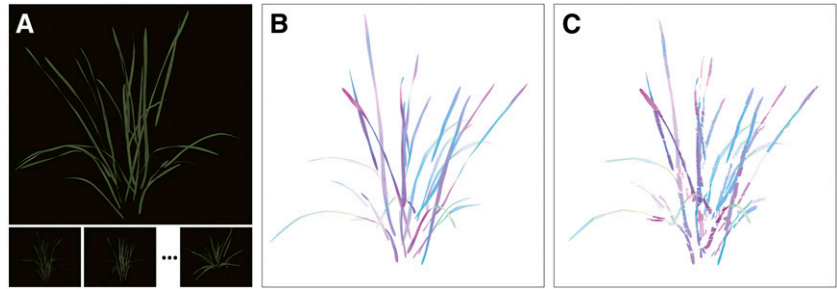


Figure 2. Boundary optimization using level sets. A, An initial surface estimate on the rice data set. B, The reconstructed rice surface after 200 iterations using our level-set approach. C, An initial surface estimate on the wheat data set. D, The reconstructed wheat surface after 200 iterations using our level-set approach. Regions showing the complex structure of the surfaces have been added above each mesh for clarity.

Figure 3. A, The model rice plant is rendered from 40 different viewpoints, moving around the plant and from above. B, The model rice plant, colored based on the surface normal orientation. C, The reconstruction of this data set using our level-set approach.



overlapping regions, something that, if not rectified, would produce errors when quantified or used in modeling. Figure 2B shows our level-set approach to have removed this overlap and smoothed each surface boundary such that the original leaf surface is recovered. This result is representative of the success of our approach across all the leaves in this data set.

The wheat data set generally contains wider leaves, and their lack of surface patterning often results in only sparse point data in the center of each leaf. This is shown in Figure 2C, where large missing sections of leaf must be recovered. Figure 2D shows our results on this section, in which the cluster boundaries have been optimized and again form a more continuous surface. There is still some overlap in the clusters on the left of the image, but this is caused by the angle from which the image was rendered and relates to the relative orientations of neighboring clusters.

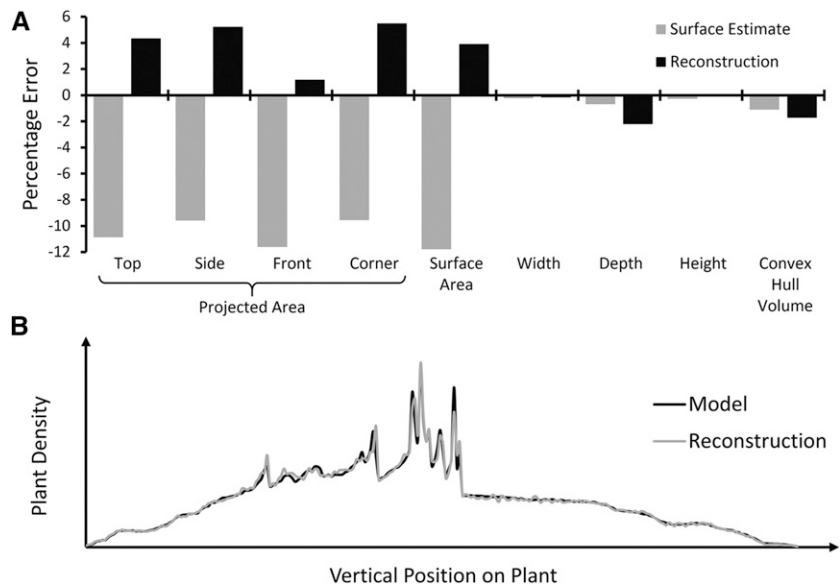
The clusters toward the left of Figure 2D are orientated at slightly disjointed angles, due to noise in the original point cloud reconstruction. This makes optimization of the intercluster boundaries challenging, as the intersection of these boundaries depends not only on their

orientation but also on the position from which they are being viewed (the reference view I_R from “Materials and Methods”). This is an important characteristic of our reconstruction algorithm in its current form: the boundaries of neighboring patches will be reshaped relative to the reference camera view I_R for each cluster. It is possible that gaps may be observable between surfaces when viewed at angles very dissimilar to the reference view. In reality, for clusters with very similar orientations, these gaps will be negligible. However, as the main focus of our reconstruction work to date has been optimizing the boundary speed functions of level sets, we have yet to address this problem. We anticipate that further work on smoothing the normal orientations of neighboring clusters or merging neighboring clusters into a single curved leaf model will solve this issue; this will be a focus of upcoming research.

Quantification of Accuracy Using in Silico Image Capture

To verify the accuracy of our reconstruction approach, an additional data set was created based on

Figure 4. Quantification of the accuracy of our approach on the virtual plant data set. A, Percentage error of various measurements between the virtual model and both the α -shape surface estimate and optimized reconstruction. Lower percentage error indicates that a measurement is closer to the original model. Projected area was measured from four viewpoints, with an orthographic projection mimicking the parallel light rays produced in our ray-tracing system. B, A graph showing the change in model density as a function of the vertical distance through the plant, from the base of the models to the top. The strong peaks represent near-horizontal leaves that cause an abundance of plant material at very specific depths.



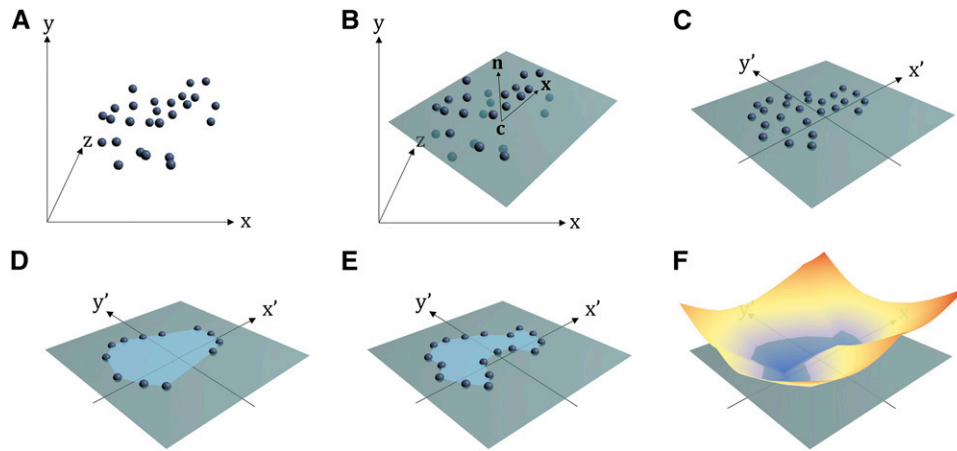


Figure 5. The process of obtaining an initial surface estimate based on a cluster of points and initializing a level-set method to optimize the surface boundary. A, A small cluster of points in 3D world coordinates, obtained through clustering the input point cloud. B, An orthogonal regression plane is fitted through the points; c , Center point of the plane; n and x , the orientation and rotation of the plane, respectively. C, Points are orthogonally projected onto the plane surface, and in doing so, the coordinate system is changed to 2D planar coordinates. The points are rotated around the normal such that the x' axis in planar coordinates lies along x . D, The boundary of the Delaunay triangulation of the 2D points in C. E, The boundary of an α -shape computed over the same point set. F, A 3D level-set function is initialized such that the intersection with the plane resembles the boundary of the α -shape computed in E.

the plant used in the rice data set. The rice plant was first manually modeled using the point cloud created by PMVS and 3D graphics software (Topogun [SC Pixelmachine SRL, version 2.0; www.topogun.com] and Blender [version 2.69; Blender Foundation; www.blender.org]). This is a time-consuming and subjective process and should not be viewed as a suitable alternative to automatic reconstruction. However, it is possible to produce an easily quantifiable ground truth model that can be used as a target for automated reconstruction. We textured and colored the virtual plant in order to emulate the original plant leaves. Finally, we rendered 40 distinct camera views of the same model (Fig. 3A), simulating an image-capture system moving around a static plant. The resulting data set can then be reconstructed in the same manner as real-world data, but importantly, we retain the ability to compare the reconstruction with the original virtual

plant model, in particular keeping the same coordinate system. The original model, and our reconstruction, can be seen in Figure 3, B and C.

As our model contains no notion of leaf or plant structure at this stage (plants are described as the combination of many distinct surfaces), we have chosen to focus our evaluation on macro-level geometric traits such as surface area instead of lower-level traits such as leaf angle or curvature. We anticipate that these additional traits will be incorporated as our methods are extended to consider topological plant structure.

A variety of geometric measures were calculated to quantify the differences between the original ground truth model, the initial surface estimate, and the model output by our reconstruction. In each case, any differences between the expected and observed results were used to produce percentage error values, which can be seen in Figure 4A. An increase or decrease in

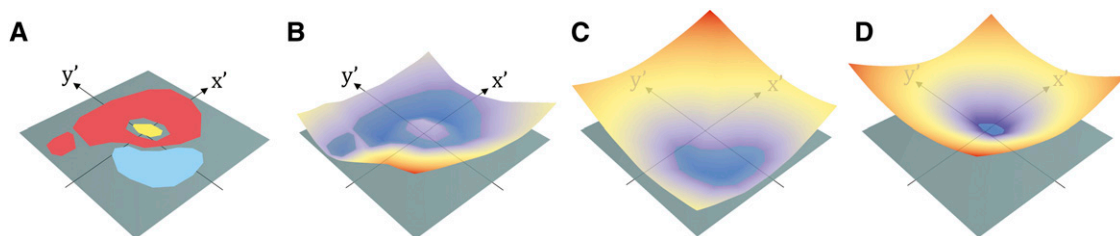


Figure 6. Level sets allow the complex boundary conditions exhibited within each leaf surface to be modeled. A, Three α -shapes in close proximity that require shape optimization. Active contours are ill suited to this task, due to the disconnected nature of the red region and the hole containing a separate cluster. B to D, Individual level-set functions intersecting the x - y plane. Using three separate level-set functions allows the complex topology of these α -shapes to be preserved and allows for efficient shape optimization.

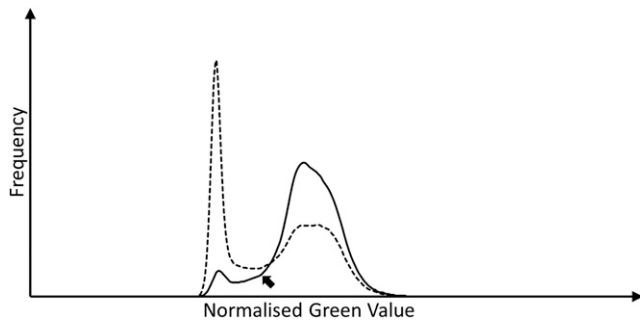


Figure 7. Histogram showing the normalized green distribution over all images in an input set. The solid line includes only locations in which the initial α -shape regions are projected, and the dashed line represents all pixels. The dashed line shows a distribution containing a larger amount of background pixels. The reduced frequency of the background in the surface-only distribution can be exploited using the unimodal thresholding method of Rosin (2001), which will select a position below the foreground peak, indicated with the arrow.

error between the initial surface estimate and our optimized reconstruction is an indication of the performance of our level-set approach.

The summary measurements of width, depth, height, and convex hull volume for the ground truth model and our reconstruction are in good agreement. The convex hull represents the smallest 3D volume that still encompasses the entire model. The results show that our approach does not significantly alter the overall size and shape of the original model. The slight decreases in depth and convex hull volume were caused by the very tip of the one leaf missing in the output model; this, in turn, could be caused by missing point data in the original cloud or the lack of a suitable reference image of that surface.

Area measurements were also calculated from both two-dimensional (2D) projected viewpoints and an overall surface area measure for each model. An orthographic projection acts in a similar way to the flattening of points onto their best-fit planes, as described above. The plant is rendered with all surfaces projected parallel directly onto the image. This removes all considerations of camera focal length and radial distortion. In all cases, the area of the reconstructed model is slightly higher than that of the original ground truth. In the worst case, the corner-projected (three-quarter) view shows a 5.5% increase in size in the reconstruction over the original model, and the total surface area was increased 3.9% from the original model. However, all area measurements show an improvement over the original surface estimate, from an average error of -10% to $+4\%$. The initial surface estimate may also contain additional flaws, such as those seen in Figure 2, that are not reflected in these geometric measurements.

We believe that the 3.9% increase in total surface area observed between the original model and our reconstruction is caused primarily by the nature of our patch-based model. As we model the curved surfaces of leaves using a series of small, flat patches, this

approximation will often differ slightly from the underlying plant (Supplemental Fig. S1). Results can be improved, to a point, by altering the size of the surface patches (Supplemental Fig. S2); however, good results usually can be obtained with the default values provided by our software. It is also possible to observe an increase in accuracy with an increase in image resolution (Supplemental Fig. S3). This benefit is marginal, however, and will come at the expense of increased cost of hardware and higher computational requirements for processing. Importantly, these results show that our approach is consistent for a large range of image resolutions and patch sizes, meaning that any user should be able to obtain good results without the need to optimize numerous parameters experimentally.

While approximating a curved surface with planar sections will inevitably reduce accuracy by a small amount, small sections are easily managed, and the algorithmic complexity of the fitting problem is reduced. Our approach, therefore, is able to reconstruct more complex surfaces than many existing algorithms applied to this type of data and, crucially, is generally applicable to plants of any size or shape. Future work will look to combine this general reconstruction approach with a more specific plant model for certain species, allowing us to obtain more accurate models of larger surfaces without completely sacrificing generality.

Software Availability

The software associated with this article is open source, distributed under a Berkeley Software Distribution license. It will be distributed on SourceForge (<http://sourceforge.net>). A link to the SourceForge distribution page is available at www.cpib.ac.uk. The software is written in C# using the .NET framework, so it is currently available only for the Windows operating system.

CONCLUSION

The recovery of accurate 3D models of plants from color images is challenging. A single plant constitutes a

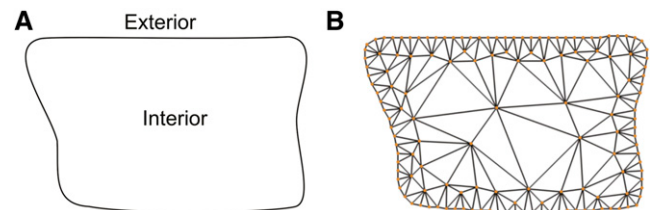


Figure 8. Constrained Delaunay triangulation of an example leaf surface. A, The level set ϕ . Input into the constrained triangulation is a list of points obtained by regularly sampling from the boundary of the level set. B, A constrained Delaunay triangulation preserves the contour of the boundary and generates additional points inside the surface to complete the mesh. [See online article for color version of this figure.]

crowded scene in the sense described by Furukawa and Ponce (2010), and the construction of accurate 3D models of objects of this level of complexity is an active research topic. Images of plants exhibit high degrees of occlusion, with the occlusion relations between leaves varying from image to image. To complicate matters further, individual leaves are difficult to identify: most of the leaves on a given plant have similar color and texture properties. Rather than address these issues in a single process that transforms a set of images into a 3D model via feature correspondence or silhouette analysis, the approach presented here develops each leaf segment individually, automatically selecting an image likely to contain the necessary information. The proposed method reduces the effect of occlusion by choosing an image with a clear view of the target surface and addresses the similarity problem by performing detailed analysis of the colors present in that image. The approach is suited to image capture with either a hand-held camera or a fixed imaging setup and will work effectively on images taken using low-cost consumer digital cameras.

The mesh representation produced provides a detailed model of the surface of the viewed plant that can be used both in modeling tasks and as a route to shoot phenotyping. It should be stressed, though, that the surface description output by the proposed technique comprises a large set of distinct planar patches rather than larger, curved surfaces describing whole leaves. The level-set method resizes and reshapes each patch to maximize its consistency with neighboring patches and the selected image, and as such the reconstructed patches provide an accurate approximation of the leaf surfaces.

Looking to the future, both field phenotyping and canopy-scale modeling will require 3D models of plant communities and canopies. The major additional challenge as the number of plants is increased is the greater incidence of occlusion between leaf surfaces. Our reconstruction algorithm operates on a best-view reference image, chosen separately for each patch. Therefore, it is robust to occlusion, as heavily obscured viewpoints are discarded. However, it is still the case that leaves that cannot be viewed clearly from at least one camera are likely to be poorly reconstructed. This makes the image acquisition process particularly important.

Further development of this technique itself will focus on the generation of more easily usable 3D models for plant phenotyping. Plant trait measurements can be obtained from a point cloud or patch-based model in a straightforward manner using existing modeling packages such as Meshlab (Cignoni et al., 2008); however, this is still a manual process in which a user can select points of interest and obtain distance, or other, measurements. Grouping neighboring patches into leaves before fitting plant-specific leaf models would remove boundary discrepancies between nearby patches, providing automatic trait measurement while remaining applicable to ray tracing. As yet, our technique

does not distinguish between plant leaves and stems. We anticipate that domain knowledge can drive the segmentation process here as well, where plant components are identified based on knowledge of expected plant structure. Finally, while the performance of our approach and that of the initial point cloud estimation appears robust to a varied number of input images and views, the determination of the optimal set of image views and mechanisms for their capture is something that should be explored.

MATERIALS AND METHODS

Growth of Plants

Cultivation of wheat (*Triticum aestivum*) plants took place in greenhouses on the Sutton Bonington campus, University of Nottingham, during the summer of 2013. Seeds of the variety Paragon were germinated on Levington's seed and modular compost (Everris). Following vernalization, they were transferred to 2-L pots with J. Arthur Bower's John Innes #2 soil-based compost. Plants were watered daily and illuminated with natural light as supplemented with 400-W high-pressure sodium (sonTAGro) bulbs (Philips). Rice (*Oryza sativa*) plants of variety IR64 were cultivated hydroponically in a controlled-environment chamber as described by Hubbart et al. (2012). Conditions were 30°C (continuous), photoperiod of 12 h, with light provided by 600-W metal halide bulbs (Philips) supplemented by domestic incandescent bulbs. Irradiance at plant height was $700 \mu\text{mol m}^{-2} \text{s}^{-1}$. Sampling took place at the leaf 11 stage.

Obtaining an Initial Point Cloud

The reconstruction algorithm described in this article uses an initial point cloud estimate as a basis for the growth of plant surfaces in three dimensions. Numerous software- and hardware-based techniques exist to obtain point clouds of objects. While we have chosen to make use of an image-based technique, PMVS (Furukawa and Ponce, 2010), in principle, other approaches, such as laser scanners or LIDAR systems (Omasa et al., 2007), also could work effectively in its place.

PMVS reconstructs a 3D point cloud model of a plant and scene based on multiple color input images. A requirement of this algorithm is that the intrinsic (focal length, etc.) and extrinsic (3D position and orientation) camera parameters be known. In the case of static-camera capture systems, such as that used in RootReader3D (Clark et al., 2011), calibration (the process of obtaining camera parameters for all images) can be performed once and the estimated parameters used until the system is reconfigured. Tools exist that can perform automatic calibration of moving camera systems, and our intention is that this software be usable with a single digital camera (we use a Canon 650D with a 35-mm lens), with images captured manually by a user. We use the VisualSFM (Wu, 2011) system to perform automatic camera calibration. This software provides a step-by-step process for camera calibration and provides an interface to the PMVS software. VisualSFM uses scale-invariant feature transform (Lowe, 1999) features to find corresponding points in pairs of images, which in turn are used to calculate the 3D position of each camera position relative to all others and relative to the model being reconstructed. There are two outputs to this process: a series of camera models, each representing the 3D position of the camera when each input image was taken, and the point cloud representation of the entire scene reconstructed by PMVS.

VisualSFM and PMVS were created to address the general image-based modeling problem and represent the current state of the art in this area. PMVS has been shown to provide high-quality descriptions of simpler, convex objects, and although plant complexity poses a challenge, it provides a sound basis for the development of plant modeling techniques.

An Initial Surface Estimate

Although each point can reasonably be expected to lie on some surface, the point cloud representation produced by PMVS contains no explicit description of those surfaces. If the final plant model is to be used with a ray-tracing system (Zhu et al., 2004; Song et al., 2013) or if details of individual leaves are required,

a surface-based description must be constructed. Methods for the reconstruction of a surface mesh from a point cloud exist (Carr et al., 2001; Kazhdan et al., 2006). Most, however, construct a single surface describing the entire point cloud. While leaves of the same plant can be considered connected in the biological sense, attempting to fit a single surface representation between multiple leaves is challenging. In fact, we wish to identify each leaf separately, to aid the reconstruction and later modeling processes. This surface identity problem is increased when extending a reconstruction approach to plant canopies. Common algorithms such as Poisson surface reconstruction (Kazhdan et al., 2006) cannot be used over complex plants or plant canopies; where they attempt to fit a single surface over the scene, they will inevitably oversimplify the complex structure.

In this work, we address this problem by representing plants as a series of small jigsaw-like surface sections, with each leaf surface being composed of multiple separate surface patches. We begin by producing an initial surface estimate based on the input point cloud before refining this estimate until the plant model is complete.

The initial surface consists of a number of small, flat surface patches (the pieces of the jigsaw puzzle) that represent small areas of plant material. Previous work in plant modeling has used planar sections to represent leaves, such as the artificially generated plant models used by Song et al. (2013). Over larger plants, or within a plant canopy, large planes represent an oversimplification of the underlying leaf shape. Smaller surface patches increase accuracy and allow neighboring sections with different orientations to characterize the curved nature of the leaves, just as smaller jigsaw pieces would allow us to better approximate the curved surface of a 3D ball puzzle.

In order to establish the required size of the fitted planes and the location and orientation of each patch, the point cloud is first segmented into small groups of points (Fig. 5A). Points are clustered with their nearest neighbors using the strategy described by Klasing et al. (2008). We extend this method to limit the potential size of each cluster, such that no surface patch becomes too large, oversimplifying the model. Points are grouped as defined by a preset distance, above which points are considered too far apart to form a cluster. This distance is dependent on the size and resolution of the model being captured. However, as the PMVS algorithm and laser scanning devices usually output points with a consistent density, the distance parameter can be set once and then remain unchanged between experiments. Here, clusters were limited to a maximum size of 150 points; reducing this number will increase the number of planar sections fitted to the data, increasing accuracy at the cost of decreased algorithmic efficiency. In our experiments, a limit of 120 points represents a significant partitioning of the point cloud, typically producing sections of leaf approximately 2 to 5 cm². Crucially, the limited size of the clusters ensures that points on neighboring leaves, between which a surface should not be formed, are rarely clustered together.

In order to approximate the surfaces throughout the model, a flat surface section is fitted to each group of points using least-squares regression. This best-fit plane minimizes the orthogonal distance to each point and is described by a center point and two vectors representing the surface orientation and rotation (Fig. 5B). It is this plane that represents the 3D position of this section of leaf surface, with the shape of the plane (which we will optimize shortly) describing the edges of that leaf surface. In performing the optimization steps on these planes, rather than the 3D point cloud, we simplify the challenging 3D surface-fitting problem into a series of less complex 2D surface-fitting problems.

Any 3D reconstruction process requires a coordinate system or frame of reference in which to describe the world. It is helpful when describing the process to visualize the different coordinate systems in use. We identify the system used by our plant model as world coordinates, representing the 3D geometry of the scene. It is in this coordinate system that the eventual completed model will be output. Each point in the input cloud is associated with the cluster of points to which it has been assigned and can be projected (flattened) onto the best-fit plane for that cluster. Once projected, we say that the point resides in 2D planar coordinates (Fig. 5C), where instead of a 3D world position, we refer to its location on the patch on which it sits. The orthographic projection also has the effect of flattening the points in each cluster to lie on their best-fit plane, reducing noise in individual points and, as mentioned above, reducing the search for an optimal patch boundary to two dimensions. It is important, however, to consider the planar and world coordinate systems as essentially different views of the same information. As such, point and mesh surfaces generated on a plane will have an associated world position that can be output as a final 3D model, and if the boundaries of our 2D patches are improved, so too is the boundary of the 3D plant model.

An initial surface description is constructed by calculating the α -shape (Edelsbrunner et al., 1983) of each set of 2D points, expressed in planar coordinates. An α -shape is a subset of the commonly used Delaunay triangulation

(which generates a triangular mesh over a set of points) but contains restrictions on which edges and faces are preserved. In general, points that are closer together will be connected by triangular faces, whereas boundary points that are farther apart will not. In practice, the value, α , can be increased in size to increase the level of detail in the boundary of the triangulation by removing larger edges. The α -value can be tuned for a given data set to preserve the shape of the boundary of each reconstructed point set. Figure 5, D and E, shows the Delaunay triangulation for the example cluster and an associated α -shape. The Delaunay triangulation oversimplifies the boundary of the shape, whereas the α -shape does not.

Shape Optimization Using Level Sets

The set of α -shapes computed over all clusters forms an initial estimate of the location and shape of the leaf surfaces. The limitations of the initial stereo reconstruction on plant data sets means that, in many instances, this estimate will be inaccurate or incomplete and will require further optimization to adequately reflect the true shape of the plant. Missing areas of leaf should be reconstructed, and overlapping shapes should be adjusted to meet at a single boundary. Many methods, such as active contours (Kass et al., 1988), parameterize the boundary of a shape before attempting this optimization: fitting a curve that is then manipulated to best describe the leaf. Such approaches are ill suited to the complex boundary conditions that might be produced by α -shapes. Consider Figure 6A, in which three example clusters have been segmented in close proximity and described by their α -shape boundaries. The red cluster contains a hole, but this hole should be preserved in order to prevent the yellow cluster from being obstructed. The red cluster also contains two distinct segments; this occurs when all the links between the two sections are longer than α and is difficult to model with a traditional contour method. This is particularly true where we can expect these two regions to merge as the optimization process continues, as is likely here.

These problems can be solved using the level-set method (Osher and Sethian, 1988; Sethian, 1999). The distinct regions and hole in the red cluster are preserved, and each region can reshape independently of the others. This is achieved by defining the boundary as the intersection of a 3D surface and a plane

The boundary of the α -shape computed for a given cluster is used to initialize a level-set function. The level-set method defines a 3D function, ϕ , that intersects the cluster plane (Fig. 6, B–D). We represent the level set as a signed distance function, where the value (the height) at any point on the function is equal to its distance from the nearest boundary. The function contains negative values within our α -shape boundary and positive values outside. Thus, the boundary itself is defined as the set of all points in ϕ that intersect the cluster plane. Consider the function in Figure 6B. It is initially shaped such that the intersection with the 2D plane matches the red region in Figure 6A. If the values of the level-set function are decreased, the 3D function will move downward and the boundary will change shape, joining up the two separate regions and eventually filling the hole in the surface. Alternatively, if the function were to move upward, the boundaries would shrink, making the surfaces continually smaller until they eventually disappear, where the 3D function no longer intersects the plane at all.

Our goal is to optimize the height of the level-set function such that the boundary represents an accurate approximation of the shape of the leaf on which the cluster is located. To achieve this, during each step of the level-set process, the height of the distance function at any point (x, y) is altered based on a speed function v . The speed function can be altered based on both global and local parameters; thus, different regions of ϕ can move upward and others downward. The result is a boundary that grows or shrinks as necessary to fit the underlying data. We have defined our speed function to consider three components: the curvature of the boundary, the underlying image data, and the interaction between neighboring surfaces:

$$v = v_{\text{curve}} + v_{\text{image}} + v_{\text{inter}}$$

Boundary Curvature

v_{curve} is a measure of the local curvature of the level set, calculated as described (Sethian, 1999). We calculate this as:

$$v_{\text{curve}} = \omega \cdot \kappa$$

where κ is the curvature at a given point on the level set, which represents the smoothness at that point, and ω is a small scaling factor used to ensure that curvature does not dictate the movement of the patch boundary compared

with v_{image} and v_{inter} . The curvature term encourages the level-set function to remain smooth, meaning that the boundary of the surfaces will also remain smooth. With any triangulation such as α -shapes, it is possible for the initial patch boundaries to be jagged (Fig. 2), while it is likely that the optimum leaf reconstruction will be smoother. Strongly convex or concave regions will be given negative or positive curvature terms, respectively, and will generally become flatter over time. An example showing the effect of this component of the speed function is shown in Supplemental Figure S4.

Image Information

The image term, v_{image} , references color information in the input images to ascertain whether the reconstructed surface lies over regions depicting plant material. We view each small patch of leaf surface from the point of view of one camera position and compare this projection with the captured image. Where an image does not show plant material at the expected location, it is likely that the surface should be shrunk at this point. Conversely, if pixels near the current surface boundary show a high likelihood of being plant material, it may be beneficial to grow the surface into this position.

While it is mathematically possible to view a given surface in any input image, it may not be useful to do so in all cases. Some images will present an obscured view of a specific leaf surface, and others may be viewed from a near-perpendicular angle, resulting in a view of a region that is less useful than other camera views. In the worst case, some cameras will not see a given surface at all, where the leaf falls outside of that camera's field of view. To avoid these difficulties, rather than combining information from multiple views, we choose instead to pick one so-called reference view in which to obtain color information. We choose a reference image that represents a calculated best view of an individual flat surface, and this is performed for each surface patch in the reconstructed model.

Reference image selection begins by projecting each cluster into each camera view (viewing each cluster surface with respect to each camera). For each initial surface, we obtain the pixel locations in which that surface appears in a specified image. Attached to each projected position is a z-depth value that records the distance that the projected point lies from the camera's image plane. These distances can be used to sort clusters that appear at the same location in an image, which may be caused by either true occlusion between leaves or overlapping surfaces caused by errors in the input point cloud. In both cases, a heavily overlapped view of a surface is a hindrance, and this image should be considered less suitable for selection as the reference view.

It is desirable to select camera views that contain as little interference between surface patches as possible. For each image and each patch, we calculate v_{clear} , a measure of the number of pixels into which a surface projects, v_{occluded} , representing the percentage of occluded pixels belonging to that cluster, and $v_{\text{occluding}}$, representing the percentage of pixels within a certain cluster that occlude those from other clusters. v_{occluded} and $v_{\text{occluding}}$ fall between 0 and 1, and in both cases values close to 1 are preferable. v_{clear} is normalized between 0, for an image in which a surface does not appear, and 1, where that surface appears at its largest. The clear pixel count also provides a measure of the angle of incidence between a cluster and the camera plane. If a cluster is seen at an angle from a given camera, this camera will likely contain fewer projected pixels than one that sees a front-on view of that leaf surface.

For each patch, the combination of normalized clear pixel count, occlusion, and occluding percentages can be used to sort images in terms of view quality:

$$\text{view quality} = v_{\text{clear}}(1 - v_{\text{occluded}})(1 - v_{\text{occluding}})$$

A reference image, I_R , is chosen for each patch that maximizes this view-quality measure. In other words, an image is chosen such that the patch appears as large as possible in the image and with as little occlusion as possible.

When referencing pixel values in the image I_R , we use a normalized green value to measure the likelihood of leaf material existing at that location. Normalized green is robust to changes in illumination, which are frequent within a canopy where shadows are cast from other leaves. It does this by disregarding luminance and considering only hue.

Computed from the red, green, and blue color channel information, the normalized green value of a pixel is calculated as:

$$\text{normalized green} = \frac{\text{green}}{\text{red} + \text{green} + \text{blue}}$$

As long as care is taken to choose a suitable background during image capture, we can assume that normalized green values will be higher in pixels containing leaf material and lower in pixels containing background. Where

lighting conditions remain consistent over an image set, we can also assume that the distribution of normalized green values is the same over each image. The expected normalized green values of leaf material for an image set should be ascertained at the beginning of the reconstruction process, before being used to contribute to the v_{image} term.

In a data set where the only strong normalized green response originates from plant material and the background intensity is fairly consistent, we can assume that the histogram of normalized green values for all pixels over all images will contain two dominant peaks, shown by the dashed line in Figure 7. If we restrict sampling to only those pixels that are initially occupied by surface patches, we would expect the frequency of background pixels to be reduced dramatically, as shown by the solid line in Figure 7. The unimodal thresholding approach of Rosin (2001) is ideally suited to analyzing such a histogram and is capable of finding an appropriate threshold level below the foreground peak. Using this point, marked in Figure 7 with an arrow, the v_{image} term can be scaled such that surfaces that appear significantly green in their reference image will grow and those that do not will shrink. The v_{image} term, therefore, is calculated as:

$$v_{\text{image}} = \begin{cases} \max\left(-1, \frac{N-t}{2\sigma}\right), & N > t \\ \min\left(+1, \frac{N-t}{2\sigma}\right), & N \leq t \end{cases}$$

where N is the normalized green value of that surface location in its reference image, t is the threshold calculated using the Rosin (2001) method, and σ is the SD of the normalized green peak. This term will produce a value between -1 , where a lower normalized green value reaches the expected color of the nonplant pixels, and $+1$, where a higher value reaches the expected color of plant material. This will cause the level-set boundary to grow over areas of plant material and shrink over areas of background.

Surface Interaction

The final component of the speed function, v_{inter} , works to reshape each surface based on the location and shape of nearby patches. As each surface patch is located on an individually oriented plane, neighboring patches might have different orientations. This makes the calculations of their 3D intersections challenging and makes decisions on how to reshape these surfaces to best avoid one another difficult. For example, there may be two neighboring patches that are separate (that do not intersect) in world coordinates but from specific camera views may obscure one another. In other camera views from different angles, they may not overlap at all. We want to find the optimum boundary shape for these patches such that they are sufficiently close in world coordinates in the final model. We use the reference image I_R for each patch and examine the interactions in this 2D view. The function v_{inter} is then calculated such that any given point is penalized if it is occluded by another surface when seen from the point of view of the reference image. We define the function as:

$$v_{\text{inter}} = \begin{cases} p, & \text{occluded in } I_R \\ 0, & \text{not occluded in } I_R \end{cases}$$

where p is a small negative value such that the level-set boundary shrinks at this location. Where this occurs, we also set the value of v_{image} at this position to 0, regardless of the normalized green value. This causes the image component to be ignored where a surface is occluded. The motivation for this is that where a surface is obstructed, we cannot reliably use image information at that position. In simple terms, the v_{inter} component of the speed function will have no effect where a surface is growing separately from all others in the model. In cases where a surface is obscured by one or more others, the boundary will shrink away from these locations.

The Complete Speed Function

The individual components of the speed function are responsible for optimizing each level set, and therefore each surface boundary, in a different way. The curvature term ensures that extremely sharp boundaries are avoided: these boundary conditions are unlikely on a real leaf, and a smoother boundary is also beneficial when outputting the final 3D model. The image term references an image chosen that provides a detailed view of that surface and will grow the surface where leaf material is observed in the image. The interaction component will examine neighboring surfaces in the same reference image and will shrink patches at locations where they are obscured. The complete speed function, therefore, works to grow each surface where leaf material is

observed while preserving the smoothness of its boundary and ensuring that overlapping surfaces are avoided.

The complete speed function is used to update each position on the level-set function, and this process is performed for each surface patch. The process must be repeated until each cluster boundary has reshaped to adequately fit the underlying image data. The speed function will slow significantly as we approach this optimal shape. Where a level-set boundary no longer moves with respect to the reference image (does not alter the number of pixels it appears in), we mark this cluster as complete and discontinue level-set iterations. Any level sets that do not slow significantly will continue until a maximum number of cycles has elapsed, a parameter that can be set by the user. We typically use between 200 and 500 iterations as a compromise between computational efficiency and offering each level set adequate time to reshape. In many cases, clusters will halt naturally before this maximum has been reached.

Remeshing the Level-Set Functions

Once the level-set operation has terminated, patch shapes may bear little resemblance to their original α -shape description. Each surface, therefore, must be retriangulated in order to provide the mesh information required for a complete plant model. The α -shape method used above is less suitable for this task, as the parameter α may have a noticeable effect on the resulting boundary shape: the level-set function now has a known boundary that was not available during the original surface estimation. This can be used to drive a more accurate meshing approach that will preserve the boundary contour.

We use constrained Delaunay triangulation for this task, based on the algorithms outlined by Shewchuk (2002). A constrained triangulation will account for a complex boundary shape when producing a mesh from a series of points; however, it will not oversimplify the boundary by fitting surfaces across concave sections and can consider holes in the surface if required. For each cluster, we sample from the boundary of the level set in order to obtain a series of points that travel clockwise around the shape. A constrained triangulation is computed from these points, a process that will automatically generate additional points, where required, within the shape itself. An example mesh generated from a single level-set function is shown in Figure 8.

As each point in the new triangulation exists in planar coordinates, they can be easily back projected into world coordinates to be output in mesh format. Our software outputs the completed mesh in the standard PLY format, which is readable in all commonly used software packages and can be imported into modeling tools.

Supplemental Data

The following materials are available in the online version of this article.

Supplemental Figure S1. Possible errors observed when using a patch-based reconstruction approach.

Supplemental Figure S2. The effect of patch count and size on reconstruction accuracy.

Supplemental Figure S3. The effect of input image resolution on reconstruction accuracy.

Supplemental Figure S4. The operation of individual components of the level set speed function.

Received August 19, 2014; accepted October 7, 2014; published October 20, 2014.

LITERATURE CITED

- Alarcon VJ, Sassenrath GF (2011) Modelling cotton (*Gossypium* spp.) leaves and canopy using computer aided geometric design (CAGD). *Ecol Modell* **222**: 1951–1963
- Alenya G, Dellen B, Torras C (2011) 3D modelling of leaves from color and ToF data for robotized plant measuring. IEEE International Conference on Robotics and Automation. IEEE, Piscataway, NJ, pp 3408–3414
- Carr JC, Beatson RK, Cherrie JB, Mitchell TJ, Fright WR, McCallum BC, Evans TR (2001) Reconstruction and representation of 3D objects with radial basis functions. *In* Proceedings of the 28th Annual Conference on

- Computer Graphics and Interactive Techniques. Association of Computing Machinery, New York, pp 67–76
- Cignoni P, Corsini M, Ranzuglia G (2008) Meshlab: an open-source 3D mesh processing system. *Ercim News* **73**: 45–46
- Clark RT, MacCurdy RB, Jung JK, Shaff JE, McCouch SR, Aneshansley DJ, Kochian LV (2011) Three-dimensional root phenotyping with a novel imaging and software platform. *Plant Physiol* **156**: 455–465
- Edelsbrunner H, Kirkpatrick DG, Seidel R (1983) On the shape of a set of points in the plane. *IEEE Trans Inf Theory* **29**: 551–559
- Furukawa Y, Ponce J (2010) Accurate, dense, and robust multiview stereopsis. *IEEE Trans Pattern Anal Mach Intell* **32**: 1362–1376
- Houle D, Govindaraju DR, Omholt S (2010) Phenomics: the next challenge. *Nat Rev Genet* **11**: 855–866
- Hubbart S, Ajigboye OO, Horton P, Murchie EH (2012) The photo-protective protein PsbS exerts control over CO₂ assimilation rate in fluctuating light in rice. *Plant J* **71**: 402–412
- Kass M, Witkin A, Terzopoulos D (1988) Snakes: active contour models. *Int J Comput Vis* **1**: 321–331
- Kazhdan M, Bolitho M, Hoppe H (2006) Poisson surface reconstruction. *In* Proceedings of the Fourth Eurographics Symposium on Geometry Processing. Association of Computing Machinery, New York, pp 61–70
- Klasing K, Wollherr D, Buss M (2008) A clustering method for efficient segmentation of 3D laser data. IEEE International Conference on Robotics and Automation. IEEE, Piscataway, NJ, pp 4043–4048
- Kumar P, Cai J, Miklavcic S (2012) High-throughput 3D modelling of plants for phenotypic analysis. *In* Proceedings of the 27th Conference on Image and Vision Computing New Zealand. Association of Computing Machinery, New York, pp 301–306
- Long SP, Zhu XG, Naidu SL, Ort DR (2006) Can improvement in photosynthesis increase crop yields? *Plant Cell Environ* **29**: 315–330
- Lowe DG (1999) Object recognition from local scale-invariant features. *In* Proceedings of the Seventh IEEE International Conference on Computer Vision. IEEE, Piscataway, NJ, pp 1150–1157
- Murchie EH, Pinto M, Horton P (2009) Agriculture and the new challenges for photosynthesis research. *New Phytol* **181**: 532–552
- Murchie EH, Reynolds M (2012) Crop radiation capture and use efficiency. *In* Encyclopaedia of Sustainability Science and Technology. Springer Science and Business Media. Springer Science and Business Media, New York, NY, pp 2615–2638
- Omasi K, Hosoi F, Konishi A (2007) 3D lidar imaging for detecting and understanding plant responses and canopy structure. *J Exp Bot* **58**: 881–898
- Osher S, Sethian JA (1988) Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. *J Comput Phys* **79**: 12–49
- Quan L, Tan P, Zeng G, Yuan L, Wang J, Kang SB (2006) Image-based plant modeling. *ACM Transactions on Graphics* **25**: 599–604
- Reynolds M, Foulkes J, Furbank R, Griffiths S, King J, Murchie E, Parry M, Slafer G (2012) Achieving yield gains in wheat. *Plant Cell Environ* **35**: 1799–1823
- Rosin PL (2001) Unimodal thresholding. *Pattern Recognit* **34**: 2083–2096
- Sethian JA (1999) Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science, Vol 3. Cambridge University Press, Cambridge, UK
- Shewchuk JR (2002) Delaunay refinement algorithms for triangular mesh generation. *Computational Geometry: Theory and Applications* **22**: 21–74
- Song Q, Zhang G, Zhu X (2013) Optimal crop canopy architecture to maximise canopy photosynthetic CO₂ uptake under elevated CO₂: a theoretical study using a mechanistic model of canopy photosynthesis. *Funct Plant Biol* **40**: 108–124
- Watanabe T, Hanan JS, Room PM, Hasegawa T, Nakagawa H, Takahashi W (2005) Rice morphogenesis and plant architecture: measurement, specification and the reconstruction of structural development by 3D architectural modelling. *Ann Bot (Lond)* **95**: 1131–1143
- White JW, Andrade-Sanchez P, Gore MA, Bronson KF, Coffelt TA, Conley MM, Feldmann KA (2012) Field-based phenomics for plant genetics research. *Field Crops Res* **133**: 101–112
- Wu C (2011) VisualSFM: a visual structure from motion system. <http://ccwu.me/vsfm/> (June 11, 2014)
- Zhong X, Peng S, Sheehy JE, Visperas RM, Liu H (2002) Relationship between tillering and leaf area index: quantifying critical leaf area index for tillering in rice. *J Agric Sci* **138**: 269–279
- Zhu XG, Ort DR, Whitmarsh J, Long SP (2004) The slow reversibility of photosystem II thermal energy dissipation on transfer from high to low light may cause large losses in carbon gain by crop canopies: a theoretical analysis. *J Exp Bot* **55**: 1167–1175