

Detailed Reconstruction of 3D Plant Root Shape ^{*}

Ying Zheng [†] Steve Gu [‡] Herbert Edelsbrunner [§] Carlo Tomasi [¶] Philip Benfey ^{||}

Abstract

We study the 3D reconstruction of plant roots from multiple 2D images. To meet the challenge caused by the delicate nature of thin branches, we make three innovations to cope with the sensitivity to image quality and calibration. First, we model the background as a harmonic function to improve the segmentation of the root in each 2D image. Second, we develop the concept of the regularized visual hull which reduces the effect of jittering and refraction by ensuring consistency with one 2D image. Third, we guarantee connectiveness through adjustments to the 3D reconstruction that minimize global error. Our software is part of a biological phenotype/genotype study of agricultural root systems. It has been tested on more than 40 plant roots and results are promising in terms of reconstruction quality and efficiency.

1. Introduction

As the primary site of nutrient and water uptake, roots play a critical role in plant growth. Recent research [15, 22] highlights the role of genes in regulating root branching, a key component of overall root architecture. A better understanding of root architecture could lead to the production of plants that sequester larger amounts of carbon dioxide, thus helping to reduce one of the causes of climate change. In addition, improved root systems can aid in food production particularly in marginal soils.

To better understand roots, it is important to be able to compare the complex 3D structure of root systems between plants with different genotypes. In contrast to simple shapes of large volume, plant roots have delicate, fine geometric structures with thin branches; see Figures 1 and 2 for the plant root imaging system and a sample image. This poses challenges for the image-based 3D reconstruction, which is exacerbated by the inaccuracies caused by unavoidable

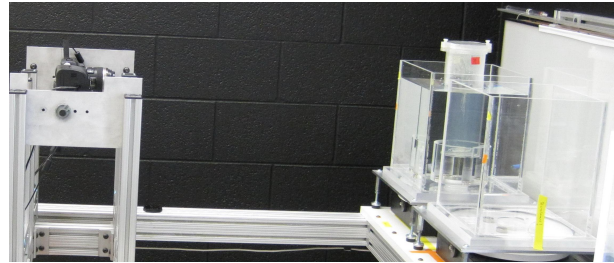


Figure 1. Plant root imaging system.



Figure 2. Close up image of two roots growing side by side in a gel container.

small refractions and the jittering inherent in the imaging system. Furthermore, there are requirements that originate from the embedding of the software in a larger work process, which include the need to have connected 3D reconstructions and software that is efficient and works without user intervention. A sample 3D reconstruction is shown in Figure 3 and additional results can be seen in Figure 9.

We make three main technical innovations to achieve the detailed 3D reconstruction of plant roots. First, we model the background of each 2D image as a harmonic function, which facilitates the extraction of the silhouette by adaptive thresholding. Second, we formulate the 3D reconstruction

^{*}This research is supported by the National Science Foundation (NSF) under grant DBI-0820624.

[†]Dept. of Computer Science, Duke Univ., Email: yuanqi@cs.duke.edu

[‡]Dept. of Computer Science, Duke Univ., Email: steve@cs.duke.edu

[§]IST Austria, Email: edels@ist.ac.at

[¶]Dept. of Computer Science, Duke Univ., Email: tomasi@cs.duke.edu

^{||}Dept. of Biology, Duke Univ., Email: philip.benfey@duke.edu



Figure 3. Five views of the reconstruction of a pair of root systems growing in a common container. Here and in the rest of the paper, the color corresponds to the height on the root.

step as a compromise between two objectives: satisfying all images and one particular image. The former objective guarantees for a good global approximation and corresponds to the traditional visual hull algorithm. Adding the latter objective, we call this the *regularized visual hull* algorithm, which reconstructs otherwise lost delicate structures. Third, we develop an algorithm inspired by persistent homology [5] that guarantees the connectedness of the 3D reconstruction. Our algorithm is efficient and runs fast in practice. For example, given a set of forty images, each consisting of $1,600 \times 1,200$ pixels, we can reconstruct the 3D root structure in seconds on a dual core laptop with only 2 GB memory.

This paper is organized as follows: Section 2 reviews prior work and explains why our problem has not been well addressed in the literature. Section 3 presents a method for extracting the binary silhouette using harmonic background subtraction. Section 4 describes the regularized visual hull that follows two optimization criteria. Section 5 presents an algorithm for ensuring the 3D reconstruction is connected. Section 6 shows and compares results obtained with our software. Section 7 concludes the paper.

2. Literature Review

The problem of reconstructing a 3D shape from 2D images has been studied for decades. The general purpose algorithm referred to as *visual hull*, or *volumetric carving*, finds the largest shape consistent with the input silhouettes or color images [1, 4, 8, 9, 13, 14, 19, 20, 10]. However, due to its sensitivity to calibration errors, thin features of the shape are likely to be lost. A joint optimization approach [7] has been proposed to cope with the segmentation and

calibration errors in the moving camera environment. It is similar to our regularized visual hull but different because it relies on the texture and color information as matching cues, which are not available in our setting. A new imaging system working with coplanar shadowgrams has been introduced in [24], in which the object and the camera remain still while the light source moves. This reduces the complexity in the calibration step from six degrees of freedom (position and orientation of the camera) to three (position of the light source), and leads to improved reconstruction results. While this method is promising, it cannot be applied in our lab setting in which the opacity of the gel poses challenges to collecting the root shadows.

Complementing the general purpose methods, there has recently been progress using prior knowledge on the shape to be reconstructed. In [6, 11, 17], shapes are reconstructed by optimizing objectives that guarantee a continuous and if possible smooth surface. However, these methods assume accurate calibration and cannot deal with jittering or other movements during the image process. Moreover, these methods are not designed for thin and delicate shapes such as plant roots. Model-based reconstruction of shapes in a restricted class, such as trees, buildings, and human bodies, has also been studied in the past decade [12, 16, 18, 21, 23]. Among this work, image-based tree modeling is the most relevant to our problem. However, this work is geared toward computer graphics applications and aims for trees that look realistic as opposed to being accurate. In particular, fine details are typically not reconstructed but instead artificially generated and added to the reconstruction. In contrast, we consider plant roots for biological studies and therefore aim at a reconstruction that is faithful to the image

data and contains as many of the fine details as possible.

To the best of our knowledge, reconstructing delicate shapes and plant roots in particular makes our problem unique. The remainder of the paper describes the novel aspects of our 3D root reconstruction algorithm as well as experimental results that provide evidence for its efficacy.

3. Harmonic Background Subtraction

We model an image as a function of intensities, $J : \Omega \rightarrow [0, 255]$, where Ω is the image grid. Assuming it represents a root growing in gel, we define the root as the *foreground* and the rest of the image as the *background*. Perhaps the simplest way to separate foreground from background is by splitting the pixels with a single intensity threshold. However, there are drawbacks because the intensity can vary from image to image as well as from one location within an image to another. We therefore propose to work with the *normalized intensity*, $I : \Omega \rightarrow [0, 1]$, defined by

$$I(x, y) = \frac{\sum_{i=0}^{J(x,y)} h[i]}{\sum_{i=0}^{255} h[i]}, \quad (1)$$

where $h[i]$ is the number of pixels with intensity i ; compare the first two pictures in Figure 6. In the rest of the paper, when we refer to an *image*, we will mean the normalized intensity function, and we will treat this function as the input to our algorithm.

We find that constructing the foreground with a single threshold can cause significant branch loss, as shown in Figure 6, in the middle. We also experiment with hysteresis thresholding [2], which works by applying a first threshold to find the main portion of the foreground and then expanding the foreground until a second threshold is reached. This generally improves the quality of the result, as shown in Figure 6, second picture from the right. Note, however, that some important fine branches are still missing.

Although the gel medium appears to be non-uniform, we observe that the values vary smoothly over the background and contain no obvious local extrema in the interior. We therefore decide to approximate the background by a harmonic function $B : \Omega \rightarrow [0, 1]$. To compute this function, we set $B(x, y) = I(x, y)$ on the boundary and enforce $\Delta B = \frac{\partial^2 B}{\partial x^2} + \frac{\partial^2 B}{\partial y^2} = 0$ in the interior of Ω . In other words, we define the background function by solving the Laplace equation with a Dirichlet boundary condition:

$$B|_{\partial\Omega} = I|_{\partial\Omega}, \quad (2)$$

$$\Delta B|_{\Omega-\partial\Omega} = 0, \quad (3)$$

where $\partial\Omega$ is the boundary of the domain. Numerically, this partial differential equation with boundary conditions can be solved using the finite element method. The right picture in Figure 4 illustrates the method by showing the harmonic background of the root image to its left.

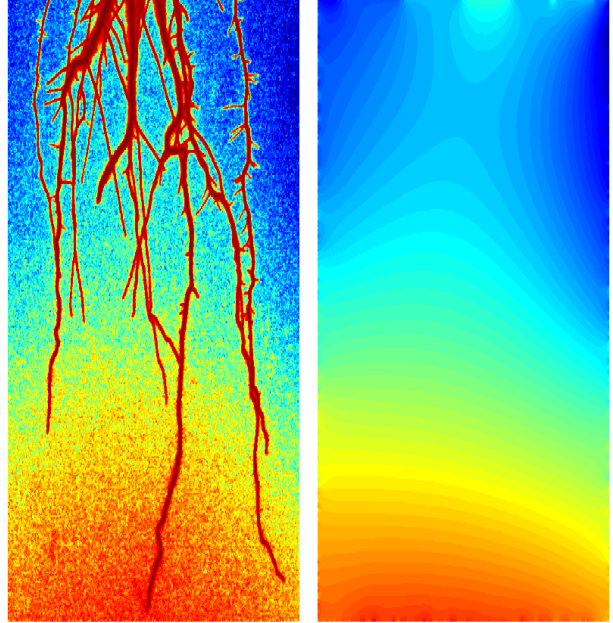


Figure 4. The (normalized) intensity of the image, I , on the left, and its harmonic background model, B , on the right.

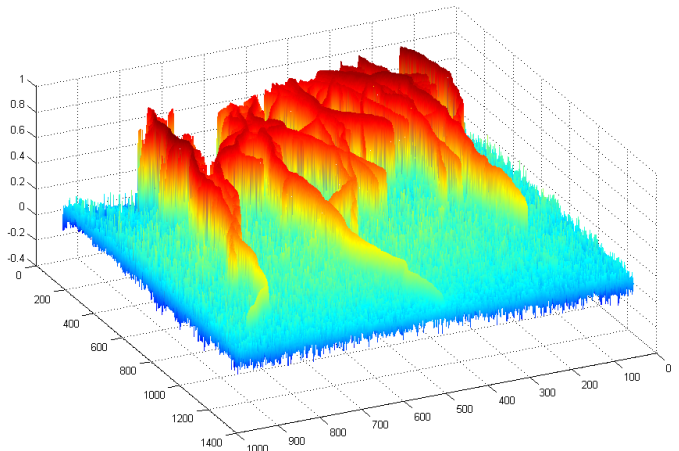


Figure 5. The difference between the normalized and the background intensity functions, $I - B$.

To construct the foreground, we use the difference between the intensity of the image and its background. As we can see in Figure 5, the foreground is greatly enhanced, so that applying hysteresis thresholding results in a qualitatively improved foreground, as shown in Figure 6, on the right.

4. Regularized Visual Hull

Typically, 3D shapes are reconstructed from foregrounds by the visual hull method. Let $I_k : \Omega_k \rightarrow [0, 1]$ be the k -

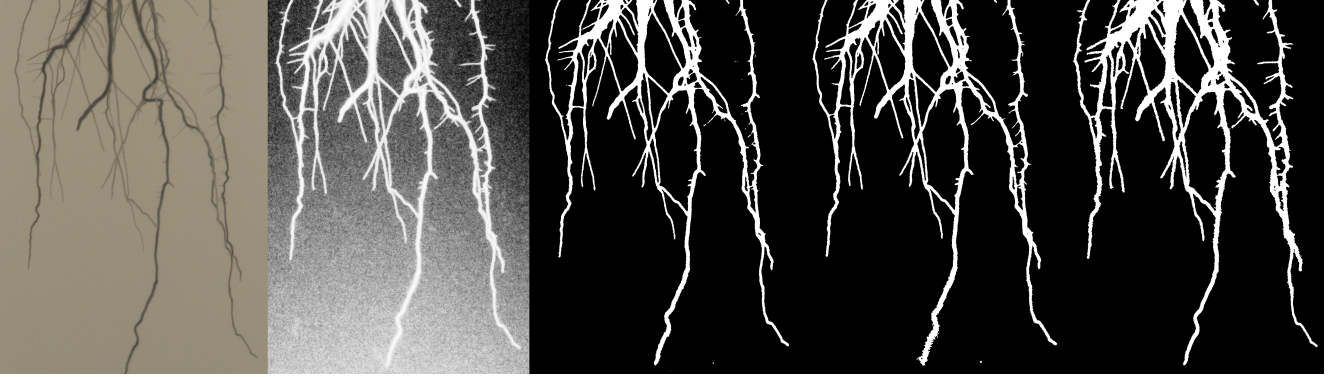


Figure 6. Images of a root system. From left to right: intensity, normalized intensity, foreground constructed by single thresholding, by hysteresis thresholding, and by harmonic background subtraction.

th image of a single plant root, for $k = 1, 2, \dots, N$. For a set V of voxels in 3D, let $\pi_k(V) \subseteq \Omega_k$ be its projection to a set of pixels in the k -th image. We write $\mathcal{F}_k \subseteq \Omega_k$ for the foreground, noting that $\pi_k^{-1}(\mathcal{F}_k)$ is the maximal set of voxels with projection \mathcal{F}_k . With this notation, we can define the *visual hull* as the maximal set of voxels whose projections are contained in all foregrounds:

$$V = \bigcap_{k=1}^N \pi_k^{-1}(\mathcal{F}_k). \quad (4)$$

Alternatively, we can describe it as the result of an optimization problem. Define the *consistency* of a voxel v with the k -th image as

$$\text{cons}_k(v) = \begin{cases} 1 & \text{if } v \in \pi_k^{-1}(\mathcal{F}_k) \\ -N & \text{otherwise,} \end{cases} \quad (5)$$

and its *total consistency* as $\text{cons}(v) = \sum_{k=1}^N \text{cons}_k(v)$. Then the visual hull is the set of voxels that maximizes the total consistency:

$$V = \arg \max_S \sum_{v \in S} \text{cons}(v). \quad (6)$$

It is not difficult to see that the two views of the visual hull are equivalent. To illustrate why the above optimization criterion is not sufficient for our purposes, we use twenty images to reconstruct the root, and assume that most of the images give good quality foreground constructions, as suggested in Figure 7. Nevertheless, even tiny distortions can cause inconsistencies between the images such that the back-projection to 3D is nearly empty. In the end, the visual hull does not match any of the input images. We suggest to use one of the twenty images to improve the 3D reconstruction. Our approach is best cast in the optimization framework with an additional regularization term. Given a set of

images, one distinguished image I_j in this set, and a *regularization parameter* $\lambda \geq 0$, the *regularized visual hull* is the set of voxels, V_λ , such that

$$V_\lambda = \arg \max_S \left\{ \sum_{v \in S} \text{cons}(v) + \lambda \cdot |\pi_j(S) \cap \mathcal{F}_j| \right\}, \quad (7)$$

where $|\cdot|$ denotes cardinality.

Note that we propose to use only one image for regularization. The reason is that jittering causes different images to contradict each other, so that using two or more images can result in duplications of the same branch. The limitation to only one distinguished image is not serious since roots are typically thin and cause only a small amount of occlusion. The regularization term may cause more voxels to be added to the solution, but it does not exclude any voxels of the visual hull. It follows that regularized visual hull induces a nested set sequence:

$$V \subseteq V_\lambda \subseteq V_\kappa, \quad \text{for all } \kappa \geq \lambda \geq 0. \quad (8)$$

We will make use of this observation when we discuss an efficient algorithm for constructing a regularized visual hull.

We now analyze the role of the regularization term and the regularization parameter, λ . Clearly, regularization encourages the covering of the distinguished foreground, \mathcal{F}_j . In other words, the new framework introduces an explicit mechanism to use one of the images to guide the 3D reconstruction. If λ is small, the distinguished image is not important and the regularized visual hull will barely differ from the visual hull. On the other hand, by choosing λ large, we can ensure that each pixel in \mathcal{F}_j is covered.

The computation of the regularized visual hull is not difficult. Using the subset relationship expressed in (8), we initialize the regularized visual hull to the visual hull: $V_\lambda = V$. Next, we visit each pixel u in \mathcal{F}_j . If u is not covered, we look for a voxel with maximal consistency measure in the

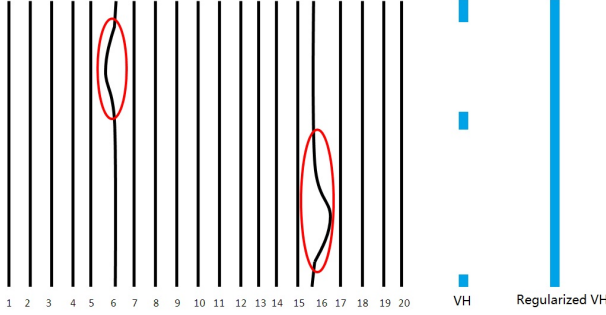


Figure 7. Left: twenty stylized root images of which two are distorted. Right: the visual hull and the regularized visual hull obtained using the first image for improvement.

set $\pi_j^{-1}(u)$:

$$v = \arg \max_{v \in \pi_j^{-1}(u)} \text{cons}(v). \quad (9)$$

Note that $\text{cons}(v)$ is negative, else u would already be covered. We then compute the regularized measure, $\text{cons}(v) + \lambda$, and add v to V_λ if that measure is positive. Otherwise, we discard v .

It is easy to prove the correctness of the above algorithm. The crucial step is to understand the role of equation (9). If v is included in V_λ , no other voxels in the set $\pi_j^{-1}(u)$ will be included, simply because its inclusion would decrease the global consistency measure while contributing nothing to the regularization term. Hence, the regularized visual hull add the minimal number of voxels to cover the distinguished image.

5. Repairing Connectivity

The regularized visual hull can consist of more than one connected component. However, for downstream applications, connectedness of the reconstruction is sometimes required, and we will see that it not difficult to be achieved. We restrict ourselves to adding voxels to the regularized visual hull, as opposed to removing voxels from it. When we add a voxel, we prefer those with low *inconsistency* with the 2D images and with small *distance* to the regularized visual hull. For each voxel v , we therefore define

$$\text{incons}(v) = \max\{-\text{cons}(v), 0\}, \quad (10)$$

$$\text{dist}(v) = \min_{w \in V_\lambda} \|v - w\|, \quad (11)$$

We can now formulate an optimization problem: find a connected set of voxels U , with $V_\lambda \subseteq U$, that minimizes the following two measures in sequence:

1. the maximum distance to V_λ ,
2. the minimum inconsistency with the 2D images.

Algorithm 1 Topology repair

```

Let  $V_\lambda$  and  $S$  be given and set  $C = S$ ;
Compute the minimal spanning tree  $\mathcal{T}$  of  $S$ ;
for each leaf node  $u$  of  $\mathcal{T}$  do
  while  $u$  is a leaf and  $u \notin V_\lambda$  do
     $C \leftarrow C \setminus \{u\}$ ;
     $u \leftarrow$  the parent of  $u$ ;
  end while
end for

```

To be specific, we use the Euclidean distance between the centers of two voxels to measure their distance, and we say two voxels are *neighbors* if they share a 2-dimensional face. A *path* is then a sequence of voxels in which any two contiguous voxels are neighbors, and U is *connected* if any two of its voxels have a connecting path within U . Similar notions of distance and connectivity are possible and lead to similar results.

We need some notation to describe an algorithm for this optimization problem. Let $d \geq 0$ be the smallest threshold such that the set of voxels $S = S_d$ with distance at most d from V_λ is connected. We optimize the first criterion by computing S with breadth-first search and limiting U to be a subset of S . By definition, $\text{incons}(v) = 0$ if $v \in V_\lambda$, and by construction, $\text{incons}(v) > 0$ if $v \in S - V_\lambda$. Note that S defines a graph in which the voxels are the *nodes* and pairs of neighboring voxels are the *edges*. We define the *weight* of an edge as the larger inconsistency of its two nodes.

Next, we compute the minimum spanning tree of this graph, noting that there are many efficient algorithms described in the literature. In this tree, there is a unique path between any two voxels, namely a *minimum cost path* that minimizes the maximum weight of its edges. We say $v \in S - V_\lambda$ *separates* if it lies on such a path connecting two voxels in V_λ . Finally, the desired solution to our optimization problem is the set U that consists of all voxels in V_λ plus all separating voxels of the minimum spanning tree. We compute U by repeatedly removing a leaf node if that node does not belong to V_λ . The algorithm stops with the desired set U . The correctness of the algorithm follows from the fact that for any two nodes in S , the minimal cost path that joins them belongs to the minimal spanning tree. After pruning the tree, we are left with all minimal cost paths that connect the components of the regularized visual hull into one component. These paths are aware of the geometry of the root structure because they achieve maximal consistency with the 2D images.

This simple algorithm is sketched in Algorithm 1. Computing the minimal spanning tree takes $O(n\alpha(n))$ time with $\alpha(n)$ the inverse Ackermann function of n and tree pruning takes only $O(n)$ time where $n = |S|$. The overall time

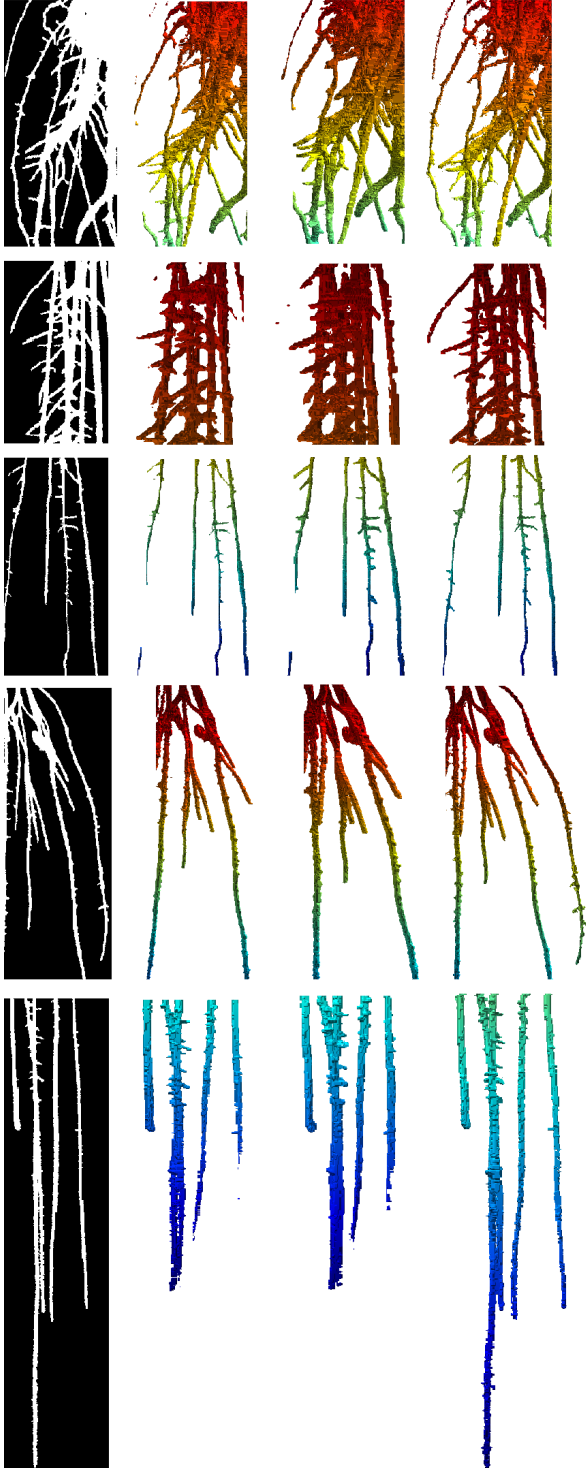


Figure 8. From left to right: the silhouette, the visual hull, an expansion of the visual hull, and the regularized visual hull after topology repair.

complexity is therefore $O(n\alpha(n))$.¹

¹Note that the MST algorithm with $O(n\alpha(n))$ time complexity is

Table 1. Comparison of visual hull (VH), the expansion of its results (eVH), and the regularized visual hull (RVH) for four different root systems.

	tp_1	fp_1	tp_2	fp_2	tp_3	fp_3	tp_4	fp_4
VH	0.87	0.00	0.93	0.00	0.89	0.00	0.97	0.00
eVH	0.90	0.58	0.98	1.36	0.97	1.06	0.99	0.54
RVH	0.92	0.03	0.95	0.01	0.94	0.02	0.98	0.01

6. Experiments

For our experimental study of the reconstruction algorithm, we reconstruct forty plant root systems growing in laboratory conditions, each described by forty 2D images taken in a circle around the plant. The root systems are grown in gel containers and vary in shape, size, and complexity. For imaging purpose, these containers are placed on top of a turntable, which is programmed to alternate between a small rotation and a stop, long enough for a single image to be acquired. The consistency of the gel allows for a small motion of the root system during the rotation, which accounts for some of the inaccuracies accumulated during data acquisition.

For camera calibration we use the orthographic projection model, although the more complicated perspective projection model is also applicable. We compare the reconstructions using our regularized visual hull algorithm with those obtained using the conventional visual hull method and with expanded versions of the latter. To quantify the results, we define two measures, called the *true positive* and the *false positive* ratios, denoted as tp and fp :

$$tp = \frac{\text{number of covered silhouette pixels}}{\text{total number of silhouette pixels}},$$

$$fp = \frac{\text{number of covered pixels not in silhouettes}}{\text{total number of silhouette pixels}}.$$

Note that tp is at most 1, while fp can be larger than 1. We choose this definition to magnify the fact that an improper 3D reconstruction can produce a large number of false positive voxels, in particular in the considered case in which the shape is thin and delicate. Also note that for the visual hull, the false positive ratio is always zero. To meaningfully compare the regularized with the conventional visual hull algorithm, we expand the reconstruction result of the visual hull result uniformly by a certain radius. The expansion recovers many of the missing voxels, but it also increases the false positive ratio. Note that $fp = 1$ means half of the back projected pixels are incorrect. The comparison of the reconstruction results using the regularized visual hull (RVH), the conventional visual hull (VH), and the expanded results of the visual hull (eVH) is given in Table 1. It confirms that

too complicated to implement. Instead, we use Kruskal's algorithm with $O(n \log(n))$ time complexity.[3, Chapter 23]

the best results are obtained with the regularized visual hull algorithm, as it increases the true positive ratio with only a very modest increase in the false positive ratio. This is nontrivial, because root structures are thin and delicate and therefore increasing fp is much easier than increasing tp .

An anecdotal visual comparison is shown in Figure 8, where we show the details of the 3D reconstruction by visual hull, the expansion of its result, and our regularized visual hull followed by topology repair. Note that in our experiments, we fixed the parameter λ to $6N$. We find that the regularization is crucial in achieving high quality results. In our experiments, the one out of the forty images that was used for improving the reconstruction was chosen randomly. We show a few representative 3D root structures reconstructed with our software in Figure 9.

7. Conclusions and Future Work

We have presented a new method for 3D plant root reconstruction. There are three major innovations in our approach. First, we model the background gel as a harmonic function and this way improves the foreground root silhouette extraction compared to conventional single or hysteresis thresholding methods. Second, we propose the regularized visual hull, which improves upon the conventional visual hull algorithm in its ability to reconstruct delicate shapes, such as thin branches of the root system. Third, we repair topological inconsistencies using minimum spanning trees.

Our software is part of a biological phenotype/genotype study of agricultural root systems. This benefits researchers in biology for their root studies. We also plan to extend our method to other delicate objects such as bones, hair, and intestines in medical imaging applications.

References

- [1] A. Broadhurst, T. Drummond, and R. Cipolla. A probabilistic framework for space carving. In *ICCV*, pages 388–393, 2001.
- [2] J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679C698, 1986.
- [3] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, 2009.
- [4] W. B. Culbertson, T. Malzbender, and G. G. Slabaugh. Generalized voxel coloring. In *Workshop on Vision Algorithms*, pages 100–115, 1999.
- [5] H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. *Discrete & Computational Geometry*, 28(4):511–533, 2002.
- [6] O. D. Faugeras and R. Keriven. Complete dense stereovision using level set methods. In *ECCV (1)*, pages 379–393, 1998.
- [7] J.-Y. Guillemot, J. Kilner, and A. Hilton. Robust graph-cut scene segmentation and reconstruction for free-viewpoint video of complex dynamic scenes. In *ICCV*, pages 809–816, 2009.
- [8] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. *International Journal of Computer Vision*, 38(3):199–218, 2000.
- [9] A. Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE Trans. Pattern Anal. Mach. Intell.*, 16(2):150–162, 1994.
- [10] S. Lazebnik, Y. Furukawa, and J. Ponce. Projective visual hulls. *International Journal of Computer Vision*, 74(2):137–165, 2007.
- [11] M. Lhuillier and L. Quan. Surface reconstruction by integrating 3d and 2d data of multiple views. In *ICCV*, pages 1313–1320, 2003.
- [12] A. R. Martinez, I. Martín, and G. Drettakis. Volumetric reconstruction and interactive rendering of trees from photographs. *ACM Trans. Graph.*, 23(3):720–727, 2004.
- [13] W. Matusik, C. Buehler, R. Raskar, S. J. Gortler, and L. McMillan. Image-based visual hulls. In *SIGGRAPH*, pages 369–374, 2000.
- [14] W. Matusik, H. Pfister, A. Ngan, P. A. Beardsley, R. Ziegler, and L. McMillan. Image-based 3d photography using opacity hulls. In *SIGGRAPH*, pages 427–437, 2002.
- [15] M. Moreno-Risueno, J. V. Norman, A. Moreno, J. Zhang, S. Ahnert, and P. Benfey. Oscillating gene expression determines competence for periodic arabidopsis root branching. *Science*, 329:1306–1311, 2010.
- [16] B. Neubert, T. Franken, and O. Deussen. Approximate image-based tree-modeling using particle flows. *ACM Trans. Graph.*, 26(3):88, 2007.
- [17] S. Paris, F. X. Sillion, and L. Quan. A surface reconstruction method using global graph cut optimization. *International Journal of Computer Vision*, 66(2):141–161, 2006.
- [18] L. Quan, P. Tan, G. Zeng, L. Yuan, J. Wang, and S. B. Kang. Image-based plant modeling. *ACM Trans. Graph.*, 25(3):599–604, 2006.
- [19] S. M. Seitz and C. R. Dyer. Photorealistic scene reconstruction by voxel coloring. *International Journal of Computer Vision*, 35(2):151–173, 1999.
- [20] G. G. Slabaugh, W. B. Culbertson, T. Malzbender, M. R. Stevens, and R. W. Schafer. Methods for volumetric reconstruction of visual scenes. *International Journal of Computer Vision*, 57(3):179–199, 2004.
- [21] P. Tan, G. Zeng, J. Wang, S. B. Kang, and L. Quan. Image-based tree modeling. *ACM Trans. Graph.*, 26(3), 2007.
- [22] J. Traas and T. Vernoux. Oscillating roots. *Science*, 329:1290–1291, 2010.
- [23] H. Xu, N. Gossett, and B. Chen. Knowledge and heuristic-based modeling of laser-scanned trees. *ACM Trans. Graph.*, 26(4), 2007.
- [24] S. Yamazaki, S. G. Narasimhan, S. Baker, and T. Kanade. Coplanar shadowgrams for acquiring visual hulls of intricate objects. In *ICCV*, pages 1–8, 2007.

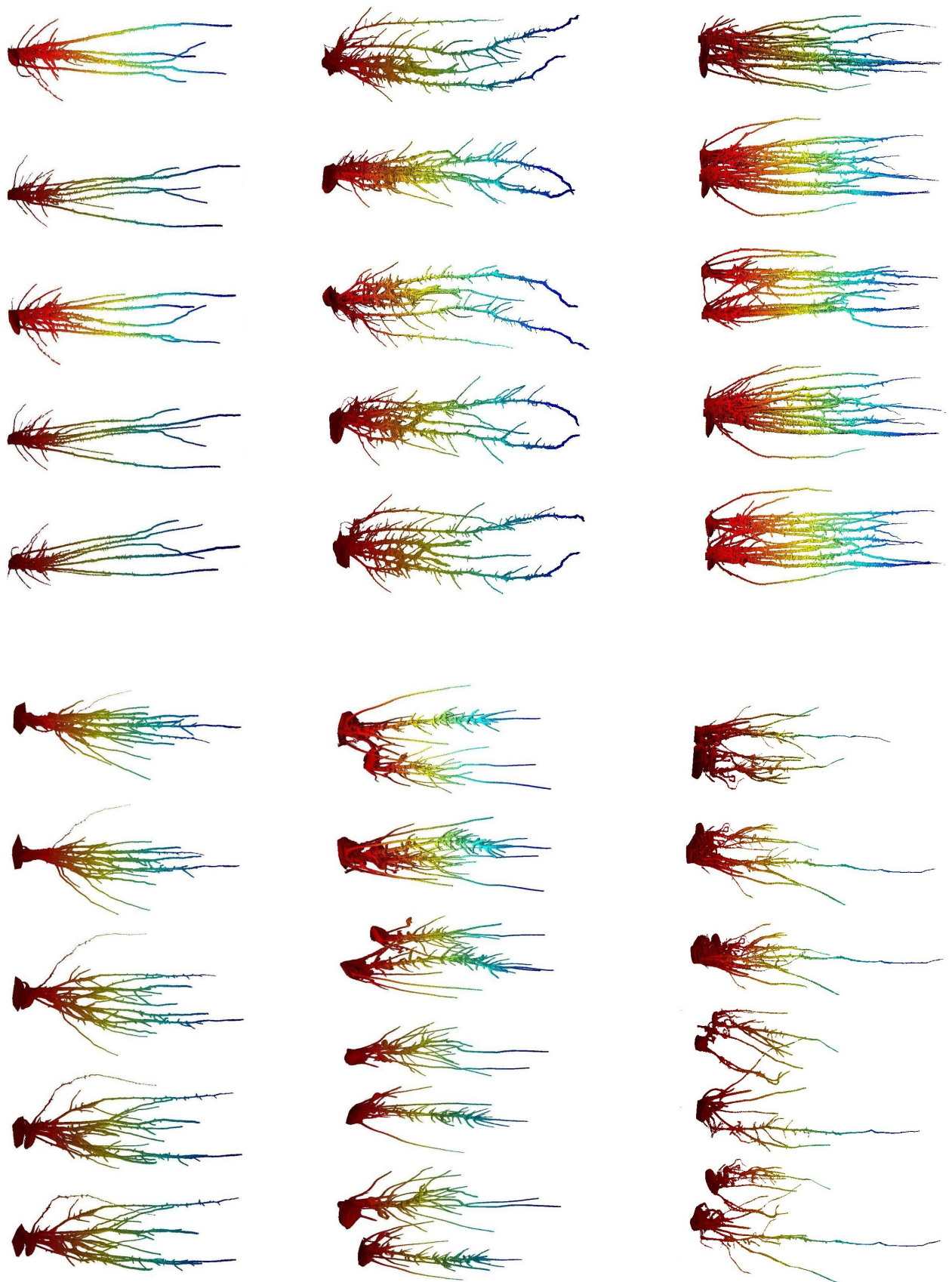


Figure 9. Six reconstructed root systems or pairs of root systems, each shown from five different directions.