

3D Shape Reconstruction of Plant Roots in a Cylindrical Tank From Multiview Images

Takeshi Masuda

National Institute of Advanced Industrial Science and Technology (AIST)

Tsukuba, Ibaraki, Japan

t.masuda@aist.go.jp

Abstract

We propose a method for reconstructing a 3D shape of live plant roots submerged in a transparent cylindrical hydroponic tank from multiple-view images for root phenotyping. The proposed method does not assume special devices and careful setups, but the geometry and material of the tank are assumed known. First, we estimate the intrinsic and extrinsic camera parameters by the SfM algorithm, and the scale and axis of the tank are estimated by chamfer matching. Second, we apply the ray tracing considering the refraction for each view, the input images are mapped to the voxels, and then multiview voxels at the same location are robustly merged to reconstruct the 3D shape. Finally, the root feature extracted from the voxel is binarized and thinned by applying the 3D Canny operator. The proposed method was applied to real a dataset, and the reconstruction results are presented.

1. Introduction

Plant phenotyping is a technology in agriculture to measure the effect of the environment to plant growth in conjunction with the genome. In the structural plant phenotyping research, plant biologists measure various traits including plant height, volumetric biomass, leaf length, and leaf area. Recent computer/device improvement enables this task fast and automatic, and high-throughput phenotyping is the current trend.

A plant is composed of above- and below-ground parts and the root phenotyping targets the latter. Since roots in soil are not visible, there are difficulties compared to the above-ground phenotyping. Direct 3D imaging systems like CT and MRI are desirable because they are applicable to live plants even planted in soil, but they have limitations of object size, portability and equipment cost [1]. The root shape in soil/water is not preserved when the plant is pulled out, and lattice frames have been developed to preserve the

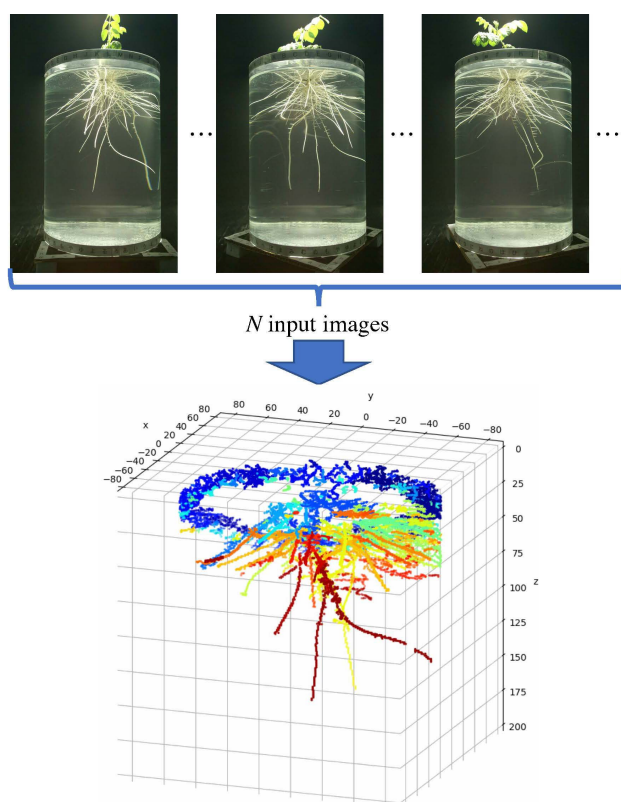


Figure 1. Input images and the extracted roots overlaid on an input image.

shape in the air [2]. However, the frame system is not perfect for keeping all details, and they are obstacles for root growth and occlusion for measurement. If the roots are grown on a thin sheet, the root system can be observed quickly by capturing/scanning as a 2D image [3, 4], but this is not a natural environment for root growth. A root phenotyping software that automatically generates root traits from a single silhouette view was developed [5]. Liu and Bucksch [6, 7] developed a system with a rotating arm with ten synchronized cameras for reconstructing 3D root struc-

ture. The target plant is maize, whose roots are intertwined very complexly, and they used 2800 images per plant. These system requires roots unearthed and dried hard, and cannot apply for measuring live plant roots.

Clark et al. [8] proposed a live 3D root phenotyping system named RootReader3D, which is based on the shape-from-silhouette method. There are several ingenious attempts in capturing images:

- The target plant is grown in a cylinder filled with transparent polymer gel. The roots are fixed static.
- For capturing images, they submerge the growth cylinder in water in a rectangular optical correction tank, which reduces the effect of refraction.
- The images are captured using a telephoto lens 2 m apart from the target, which reduces the effect of perspective, lens distortion, and refraction, and the orthogonal projection can be assumed.
- The submerged growth cylinder is rotated by a magnetically interfaced and computer-controlled rotation table. The rotation angles are reliable.
- Behind the scene, there is a lightbox, which makes silhouette extraction easy.

These settings are well organized, but it requires ample space, specialized purpose devices, and tedious work of submerging the cylinder in the optical correction tank.

Zheng et al. [9] proposed a method for 3D plant root shape reconstruction. Their settings are similar to those of Clark et al.'s [8], and their improvements are background modeling with a harmonic function for better silhouette extraction, adding a regularization term to the visual hull for robustness, and repairing connectivity concerning the consistency.

These works both assume the orthogonal projection and canceled refraction to make the problem simpler. There are several attempts to make underwater 3D reconstruction possible even in the environment with refraction. Yano et al. [10] proposed a shape-from-silhouette method of underwater objects from images captured from outside of the tank along with the simple calibration method for it. Pederesen et al. [11] proposed a camera calibration method for underwater 3D reconstruction considering refraction. For reconstructing underwater objects' 3D shapes using underwater cameras, the refraction at the interfaces of housing glass with the air and water needs to be taken into account [12, 13]. Other than these passive methods, Yamashita et al. [14] proposed a laser range finder system for underwater objects in a cylindrical glass tank.

In this paper, we propose a 3D root shape reconstruction method of a live plant which is grown in a transparent cylindrical hydroponic tank in a more in-the-wild way (Fig. 1).

We observe the roots from multiple viewpoints on the side of the tank and reconstruct the 3D root shape in the robust voxel-coloring method [15] considering refraction. Because the roots are submerged in water and the lens is not limited to telephoto, we cannot neglect the effect of refraction and perspective projection, and conventional 3D shape reconstruction algorithms based on the pin-hole camera model cannot be applied. Also, there are no distinct feature points on the roots, and feature-matching based method cannot be applied. We assume that the tank geometry and materials are known, but not the precise viewpoints and usage of specialized devices.

2. 3D Root Shape Reconstruction from Multi-view Images

In this section, we explain how to reconstruct 3D root shape reconstruction from multiview images. In Sec. 2.1, we explain how the input images are captured. In Sec. 2.2, the intrinsic and extrinsic camera parameters are estimated by the structure-from-motion algorithm, and the tank-centered world coordinate system is estimated by chamfer matching. In Sec. 2.3, the input images are projected on to the voxels, and they are merged in Sec. 2.4. Finally, the roots are extracted to be visualized in Sec. 2.5.

2.1. Setup

The plant is grown in a transparent cylindrical hydroponic tank filled with water. The tank is placed on a rotation table, which is controlled manually. The camera is fixed to look at the tank from the side of it. The camera is not necessary to be placed far away from the tank, and we need to consider the perspective projection. After each image is captured, the table is rotated by an equal amount of angle by manual control. An image capturing sequence finishes when the total rotation angle reaches 360 degrees. As a result, we obtain a set of N multiview images of the tank. We assume that the geometry and material of the tank are known.

2.2. Estimation of Geometry

2.2.1 Structure from Motion

Since the camera position relative to the tank is not known, and the rotation angle is not precisely controlled, we apply the structure from motion (SfM) algorithm to estimate them.

To make the SfM easier, **belt-shaped markers with the random pattern were attached at the top and the bottom of the exterior of the tank**. The parts between the markers are masked out because submerged parts confuse the SfM algorithm to produce wrong results with longer computation time.

As the result of the SfM algorithm, **we obtain the intrinsic parameters of the camera** (f : focal length in pix-

els; c_x, c_y : optical center; k_1, k_2, p_1, p_2, k_3 : distortion parameters), the extrinsic parameters of the views $M_l = [R_l | t_l]$ ($1 \leq l \leq N$), sparse point clouds and the undistorted input images I_l ($1 \leq l \leq N$).

2.2.2 Ray Tracing – Intersection with the Tank

Assume that the lens distortion is calibrated beforehand, and the camera projection is described by a projection matrix

$$P = \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix}.$$

For each pixel at (u, v) , the ray vector that points the pixel from the optical center is represented by $v(u, v) = (u - c_x, v - c_y, f)$. We assume that the extrinsic camera parameters are given by $[R | t]$, where R and t represent the pose and position of the camera in the world coordinate system, respectively. The ray vector and camera position in the world coordinate system is given by $v_0(u, v) = R \cdot v(u, v) / \|v(u, v)\|$ and $p_0 = t$ respectively. The equation of the ray emitted from the camera center passing through the pixel (u, v) is given by $p_0 + tv_0(u, v)$.

We lay the tank coordinate system so that the origin is at the center of the top disc of the cylindrical tank and the z -axis pointing to the bottom. When a ray $p + tv$ intersects the cylinder of radius r whose axis is passing through the origin along the z -axis, the parameter \tilde{t} at the intersection is given by

$$\begin{aligned} a &= v_x^2 + v_y^2 \\ b &= 2(p_x v_x + p_y v_y) \\ c &= p_x^2 + p_y^2 - r^2 \\ d &= b^2 - 4ac \\ \tilde{t} &= \begin{cases} (-b + s\sqrt{d})/2a & \text{if } d \geq 0, \\ \text{NaN} & \text{otherwise,} \end{cases} \end{aligned}$$

where s signifies which of the frontal ($s = -1$) and backward ($s = +1$) surface of the cylinder to the camera intersects the ray. The first intersection of the ray with the tank's exterior is determined by $p + ((-b - \sqrt{d})/2a)v$, and its z -axis coordinate is limited between 0 and the depth h .

In summary, given the intrinsic parameters P , extrinsic parameters $[R | t]$, tank geometry (radius r and height h), for each pixel, we obtain the coordinates of the intersection of the ray with the tank. For the pixels whose rays do not intersect the tank is filled with the NaN (not a number). This procedure is described by the intersection function $\mathbf{p}' = X(\mathbf{p}, \mathbf{v}, r, h, s)$.

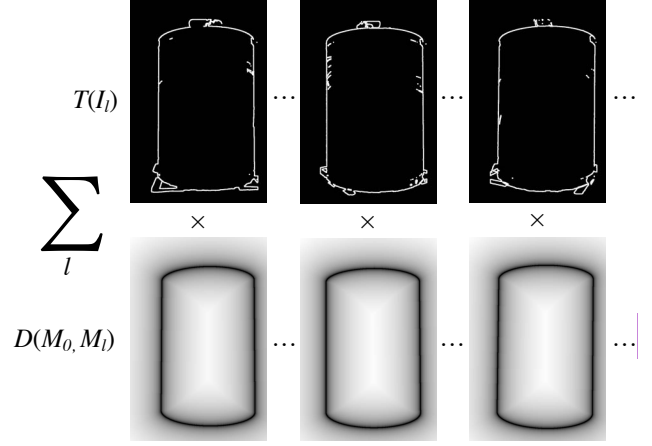


Figure 2. The chamfer loss function for the scale and axis estimation.

2.2.3 Scale and Axis Estimation

The result of the SfM in Sec. 2.2.1 does not reflect the actual object size, and its coordinate system is not tank-centered. As mentioned beforehand, the tank coordinate system is laid so that the origin is at the center of the top and the z -axis is pointing to the bottom (Fig. 4). We need to estimate the relative scaled Euclidean transform from the SfM coordinate system to the tank coordinate system.

Suppose we estimate the transform $M_0 = [s_0 R_0 | t_0]$, which is a scaled rigid transform between the SfM coordinate system to the tank coordinate system. For the l -th view ($1 \leq l \leq N$), the viewpoint and the ray vector in the tank coordinate system are represented by $M_0 t_l$ and $R_0 R_l v$ respectively. By using the intersection function defined in the previous section, we can generate the silhouette of the tank by judging whether the ray intersects the tank or not for each pixel. We formulate the problem to find the optimal M_0 that match the generated tank silhouette to the undistorted image for all views.

The loss function is derived from chamfer matching (Fig. 2). For each candidate M_0 , we generate the distance map $D(M_0, M_l)$ from the contour of the tank silhouette. For each undistorted input image, we apply the graph-cut algorithm [16] initialized by the foreground and background regions to extract the tank region. On the contour pixels of the extracted region $T(I_l)$, the distance map $D(M_0, M_l)$ is summarized. The optimization is described by

$$\tilde{M}_0 = \underset{M_0}{\operatorname{argmin}} \sum_{l=1}^N \rho(D(M_0, M_l) T(I_l)),$$

where ρ is the M-estimator to eliminate the outliers since the extracted tank contour $T(I_l)$ is not perfect.

In the implementation, for fast computation, the undistorted input images are downsampled to 1/4 of the original.

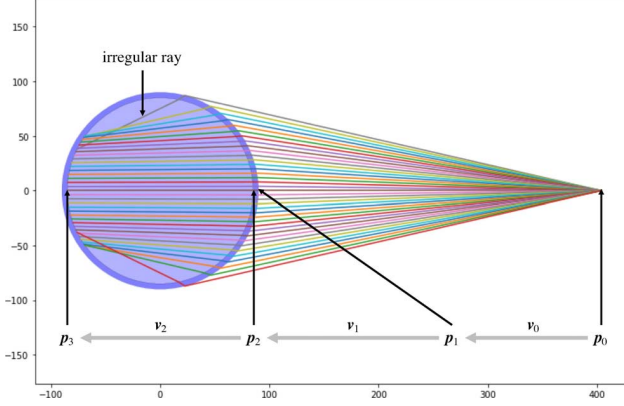


Figure 3. The ray emitted at p_0 proceeds in the air in the ray vector v_0 . It intersects the exterior of the tank at p_1 , and it is refracted to the ray vector v_1 within the tank wall. Then, it intersects the interior of the tank at p_2 . After refraction, it penetrates the water in the ray vector v_2 , and finally intersect the interior of the tank at p_3 . The rays that cause the folds of p_3 are eliminated as irregular.

Instead of using all views, for $N = 36$, we started by using only the first image with the manually given initial state, then gradually increase the number of views using the converged result as the initial state like as; with 3 of every 12 views with $\rho(x) = 2(\sqrt{1+x^2}-1)$ (Soft-L1); 6 of every 6 views with $\rho(x) = \ln(1+x^2)$ (Cauchy); 12 of every 3 views with Cauchy, etc. The minimization algorithm is Powell, and the rotation is parameterized by the angle-axis representation.

Once the scale and axis of the tank is estimated, the extrinsic parameters of the views are obtained by

$$\tilde{M}_l = \tilde{M}_0 M_l, 1 \leq l \leq N.$$

2.3. Voxel Rendering

2.3.1 Ray Tracing in the Tank

In the world coordinate system, the tank is represented by a cylinder whose radius is r , both ends are at $z = 0$ or h , and its wall thickness is τ . The ray intersection to the tank surface is modeled by the intersection function X defined in Sec. 2.2.2.

At each interface of materials, the ray is refracted under Snell's law. In our environment, there are three materials: air, tank, and water, whose refractive indices are n_1 , n_2 , and n_3 . We represent the refraction at the interface of materials n_1 and n_2 by $v' = K(p, v, n_1, n_2)$.

The path of the ray for the l -th view is traced as the followings (Fig. 3):

1. The rays start at $p_0 = \tilde{M}_l(t_l)$ in the ray vector $v_0 = \tilde{R}_0 R_l v$;
2. intersect with the tank's exterior: $p_1 = X(p_0, v_0, r, h, -1)$;

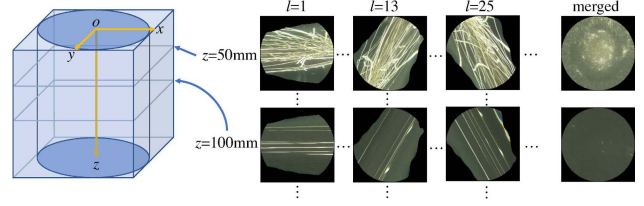


Figure 4. The coordinate system of the voxel (left). The z -slices of rendered voxels I_{ijkl} (middle). The corresponding slices of the merged voxel V_{ijk} (right).

3. refract at the tank's exterior from the air: $v_1 = K(p_1, v_0, n_1, n_2)$;
4. intersect with the tank's interior: $p_2 = X(p_1, v_1, r - \tau, h - \tau, -1)$;
5. refract at the tank's interior to the water: $v_2 = K(p_2, v_1, n_2, n_3)$.

The ray started at p_0 passed the pixel (u, v) with the ray vector $v_0(u, v)$ finally passes through the water starting at the point $p_2(u, v)$ on the tank's interior surface in the direction $v_2(u, v)$.

2.3.2 Mapping Pixels on Voxels

In the previous section, we know for each pixel (u, v) the ray passes through the line $p_2(u, v) + tv_2(u, v)$ in the water. For 3D reconstruction, we discretize the 3D space into voxels at $p_{ijk} = (i\delta, j\delta, k\delta)$, and we need to know the inverse problem: for each voxel point p_{ijk} , from which pixel (u, v) the ray comes. The closest ray from p_{ijk} is given by

$$C(p_{ijk}; p_2, v_2) = \underset{(u,v)}{\operatorname{argmin}} \min_t \|p_2(u, v) + tv_2(u, v) - p_{ijk}\|_2.$$

The direct computation of the function C for all p_{ijk} in terms of all pixels (u, v) is slow, and we apply two-step acceleration relying on the smoothness. 1) Instead of taking all (u, v) at a time, we first split the image into equal-sized square patches (e.g., 64×64) and compute the function only at the center points of the patches, and then apply the function for all pixels in the patch that gives the minimum value. 2) Instead of computing at all p , we subsample the space (e.g., in the 10-voxel interval) and apply the RBF interpolation.

As a result, for the l -th view, for each voxel point p_{ijk} , the corresponding pixel is given by $C(p_{ijk}; p_2, v_2)$, and the pixel value at $C(p_{ijk}; p_2, v_2)$ of the input image I_l is mapped to the voxels which are described by I_{ijkl} . In practice, for a specific l and k , we get a slice of the voxels as an image representing how the ray from the input image passes through the water.

2.3.3 Irregular Rays

The rays passing near the edge of the cylinder intercross the other rays which cause the spurious mapping to the images (Figs. 3 and 8). These irregular rays are detected by the fold of the p_3 , which is the second intersection of the ray with the tank's interior. The regular region is detected by

$$\det \left[\frac{\partial p_3}{\partial u} \mid \frac{\partial p_3}{\partial v} \mid N(p_3) \right] \geq 0,$$

which is the determinant of the gradients of p_3 in terms of the pixel coordinate (u, v) and the surface normal at p_3 , where

$$p_3 = X(p_2, v_2, r - \tau, h - \tau, +1),$$

and the surface normal at a 3D point p on a cylinder can easily be obtained by $N(p) = (p_x / (p_x^2 + p_y^2), p_y / (p_x^2 + p_y^2), 0)$. After the regular rays are determined, the voxels whose distance to the closest ray is within the distance δ is used for further computation.

2.4. Merging Multiview Voxels

We merge the multiview voxels I_{ijkl} to form a single voxel V_{ijk} . Conventional multiview reconstruction methods take logical-and of the binarized silhouette to form a visual hull, but this is not robust to noise like binarization failure, calibration error, and slight deformation during multiview capture.

A 3D point can be occluded by other roots and cannot be seen from all views. Instead of the binary shape-from-silhouette method, assuming that the multiview rays coincide together at a 3D point with the same pixel values, we used the geometric median that minimizes the sum of non-squared l_2 -norm. This is an L_1 estimator which is robust in a sense in the sense that its breakdown point is 0.5, and it is applicable to multi-dimensional values. In the undistorted input images, each pixel has RGB values, and it is inherited to the corresponding voxel in I_{ijkl} . For each voxel coordinates (i, j, k) , we apply the geometric median:

$$V_{ijk} = \underset{x}{\operatorname{argmin}} \sum_{l=0}^N \|x - I_{ijkl}\|_2.$$

In this computation, irregular voxels detected in Sec. 2.3.3 are excluded.

2.5. Root Extraction

In the merged voxels, the roots should have string shape with equal pixel values. To extract the roots from the merged voxel, we used the features that were used to extract vessels from CT or MRI images proposed by Sato et al. [17]. This algorithm computes the eigenvalues ($\lambda_1 \geq \lambda_2 \geq$

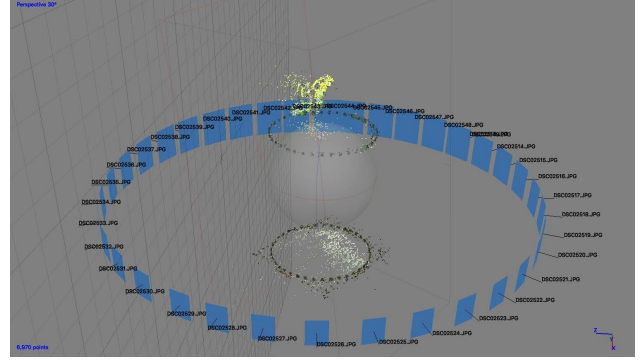


Figure 5. SfM result

λ_3) of the Hessian of the voxel values to generate feature values, and use the fact that $\lambda_1 \approx 0$ and $\lambda_2 \approx \lambda_3 \ll 0$ stand for brighter vessel-like objects in the darker background. We use the gray-scale value of each voxel for adopting this algorithm, and we call the generated feature as rootness.

For thinning the rootness, we extended Canny edge detector [18] for 3D voxels. Instead of using 2D image edge operator for edge orientation estimation, we use the eigenvector corresponding to the eigenvalue λ_1 as the orientation of root. For non-maximum suppression, the orientation of eigenvector is discretized so that its coordinates are in -1, 0, +1, and 13 dilation masks of $3 \times 3 \times 3$ are applied depending on the discretized orientation. We inherit the double thresholds for robust tracking of roots with hysteresis. The voxels with rootness higher than the high threshold are categorized as root voxels, and those higher than the low threshold are considered as root voxels if they are connected to the root voxels determined by the high threshold.

Finally, the extracted roots are segmented and remapped on the input undistorted images for easy check of the extraction results.

3. Experiments

The target plant of the experiment is licorice (*Glycyrrhiza*) and the images are offered by Kajima Corporation. Each image was captured in every about 10 degrees of rotation, and $N = 36$ images are used for the experiment. The camera model is Sony DSLR-A100 with the image resolution of 3872×2592 , and the focal length of the lens is 24 mm. The camera was placed about 40 cm apart from the tank. The tank is made of transparent Poly(methyl methacrylate) (PMMA), whose radius, height and wall thickness are $r = 90.0$ mm, $h = 246.6$ mm, and $\tau = 5$ mm respectively. The refraction index of air, PMMA and water are $n_1 = 1.000$, $n_2 = 1.490$, $n_3 = 1.333$ respectively. The lighting conditions and camera exposure were fixed throughout capturing the total views.

As the SfM algorithm, we used Agisoft's Metashape

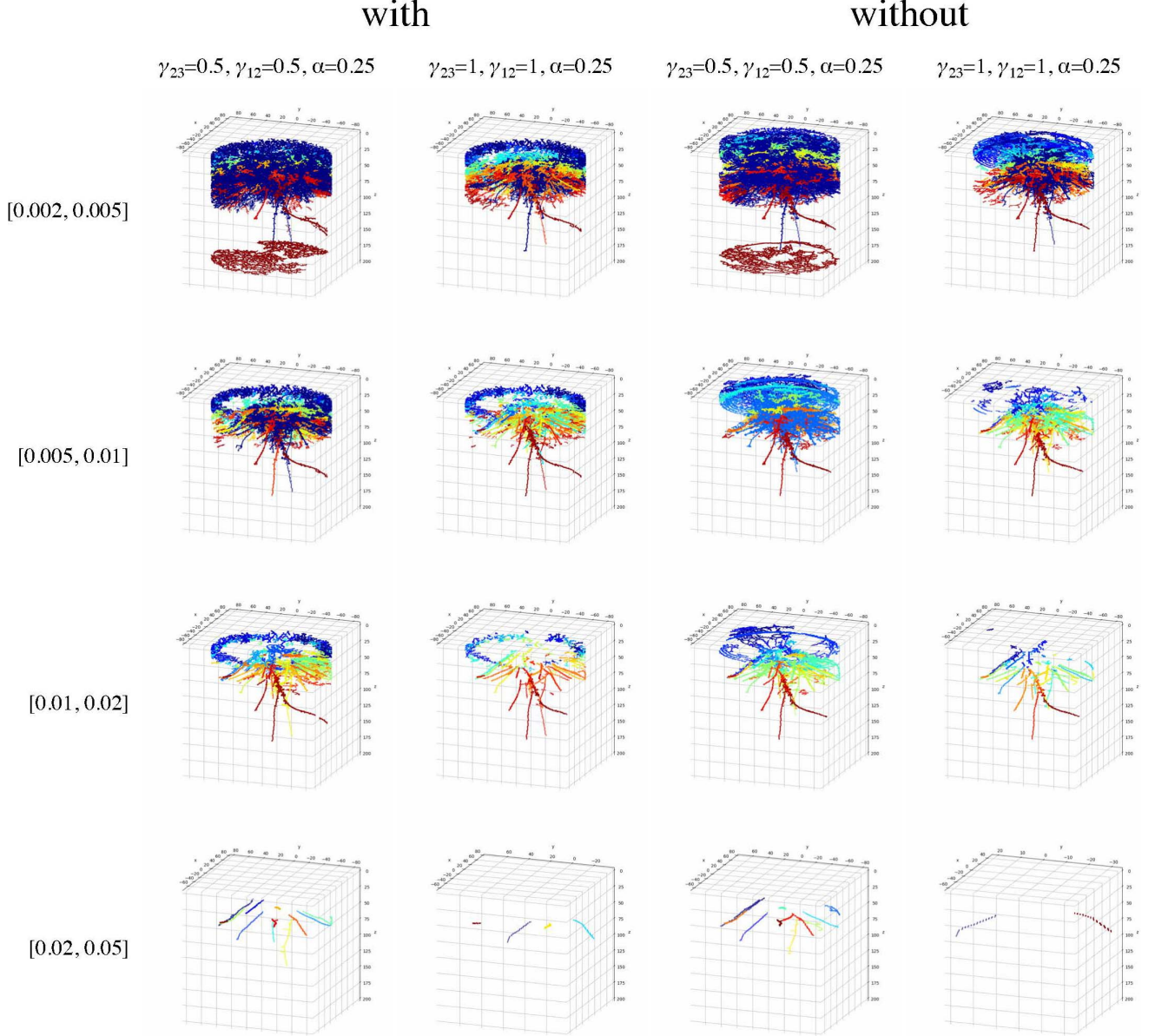


Figure 6. The reconstruction results with various parameters. The left and right two columns are the results with and without irregular ray exclusion, respectively. For each case, we tested two sets of parameters of γ_{23} , γ_{12} , and α for Sato's feature [17]. Each line corresponds to pairs of low and high thresholds for Canny operator [18] which we extended to 3D.

[19] (Fig. 5). Belt-shaped markers with random patterns were attached at both edges of the sidewall of the tank to help the SfM. The rectangular region between these markers were masked out not to confuse the SfM by non-pinhole projection due to refraction. The RMS reprojection error was 0.822 pixels. From the output of the software, we used the intrinsic (f, c_x, c_y) and extrinsic parameters $M_l (1 \leq l \leq N)$ of the camera and the undistorted input images $I_l, (1 \leq l \leq N)$ for further procedures.

As mentioned in Sec. 2.2.3, the number of views were

increased step-by-step in the optimization for the scale and axis estimation. We considered this process converged when using 12 in every 3 images, and top four digits of the estimated parameters were stabilized while incrementing the number of views.

The voxels were sampled with the interval $\delta = 1$ mm, which is the resolution of reconstruction, and we sampled up to the depth of $z = 180$ mm. The inner diameter of the tank is $2(r - \tau) = 170$ mm, and from the $N = 36$ views, we generated the voxels I_{ijkl} of size $171 \times 171 \times 180 \times 36$.

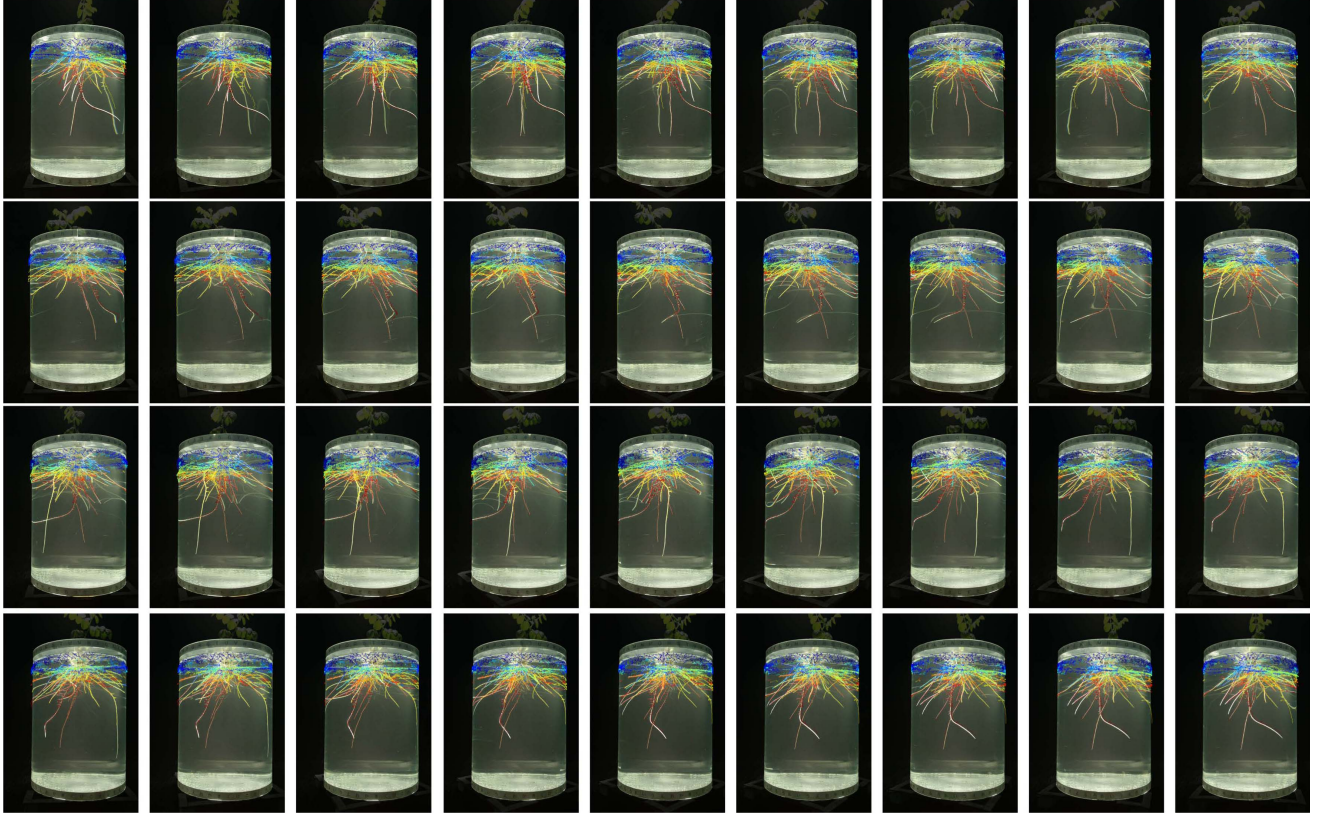


Figure 7. The extracted roots overlaid on the input images. The images are in the row-major order.

They are merged to the voxel V_{ijk} of size $171 \times 171 \times 180$.

Figure 6 shows 3D shape reconstruction results with various parameter settings. We tested two sets of parameters $\gamma_{23} = 0.5, \gamma_{12} = 0.5, \alpha = 0.25$ and $\gamma_{23} = 1, \gamma_{12} = 1, \alpha = 0.25$ of Sato's feature which are most commonly used in their paper [17] and four pairs of low and high thresholds for 3D Canny operator (Sec. 2.5). Comparing these results, the results with $\gamma_{23} = 0.5, \gamma_{12} = 0.5, \alpha = 0.25$ contains more roots than those with $\gamma_{23} = 1, \gamma_{12} = 1, \alpha = 0.25$ for the same pair of thresholds. Unfortunately, we cannot evaluate these results with the ground truth quantitatively, but we chose the setting with the irregular ray exclusion, $\gamma_{23} = 0.5, \gamma_{12} = 0.5, \alpha = 0.25$ and $[0.01, 0.02]$.

Figure 7 shows the extracted roots overlaid on the input images. The region out of the estimated tank region is darkened, which shows that the scale and axis estimation in Sec. 2.2.3 worked fine. One can see there are many illusory roots caused by reflection at the inner surface of the tank in the input images, and the extraction was successful without affected by these reflections because their appearance changes by viewpoint change differ to those of the actual roots. In the first 6 images, a reconstructed root (yellow in the figure) is not matched to the real one in the images. This is possibly due to the movement of the root caused by the

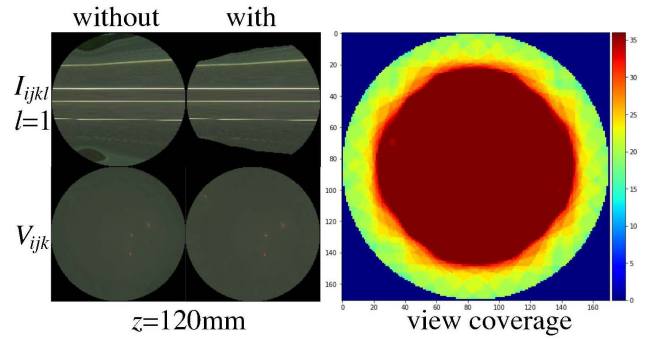


Figure 8. Effect of irregular ray exclusion and the coverage of included views at $z = 120$ mm. Red dots in V_{ijk} signify the extracted root positions in the slice.

tank rotation during the measurement, and it was successfully overcome by robust merging. There are some artifacts caused by the water surface which can be removed in post-processing.

In Sec. 2.3.3, irregular rays are excluded in merging, which is for better extraction of the roots near the tank wall in a precise comparison of the merging results with/without this exclusion (Fig. 8 left). Although the coverage of the non-excluded views is reduced nearly to $N/2$ near the tank

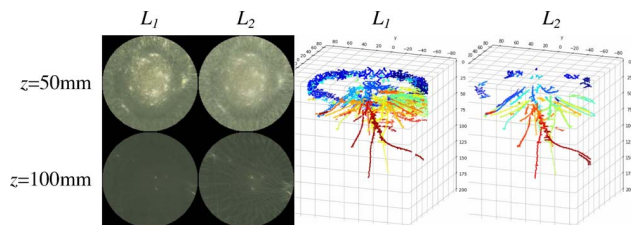


Figure 9. Comparison of the L_1 and L_2 estimator for voxel merging.

wall (Fig. 8 right), implicit exclusion of irregular regions give better reconstruction near the tank wall due to robust merging method whose breakdown point is 50 % (Sec. 2.4).

Figure 9 shows the comparison of the L_1 and L_2 estimator for multiview voxel merging. The L_1 and L_2 estimators minimize the non-squared and squared sum of the pixel value differences, respectively. The merged voxels of L_2 estimator look more blurred than those of the L_1 estimator, and the result of L_2 estimator contains less extracted roots than that of the L_1 estimator. As for the color difference metric, we also tested the CIE $L^*a^*b^*$ in addition to the RGB, but there was not such evident difference between them.

The most time-consuming part was the voxel rendering in Sec. 2.3, and it took about 12 min per view of $171 \times 171 \times 180$ voxels on 3.1 GHz CPU, which was accelerated by the multi-threading technique. The subsampling parameters (64×64 , 10-voxel interval) in Sec. 2.3.2 were determined empirically balancing the computation cost and interpolation error.

4. Conclusion

In this paper, we propose a method for 3D shape reconstruction of plant roots grown in a transparent cylindrical hydroponic tank from images observed from multiple viewpoints around it. We do not assume particular types of equipment and careful setups but assume knowledge of the tank geometry and material. The camera parameters are estimated by the SfM algorithm followed by chamfer matching of the tank silhouette, and we do not need to control the viewpoint precisely. Instead of applying binarization and the shape-from-silhouette algorithm, the input multiview images are directly mapped on the voxels aligned on the tank coordinate system, and the mapped voxels at the same location with similar pixel values are merged robustly. From the merged voxel values, we determine the rootness, and it is binarized and thinned by applying the 3D-extended Canny operator. The method overcomes the problems caused by irregular rays, voxel interpolation error, and root movement.

Future works should include computation acceleration, e.g. with GPU, more precise reconstruction, capabilities for

more difficult/complex roots, tolerance to the tank variance, more sophisticated markers for more straightforward view estimation, and learning and evaluation with simulation and the ground truth.

Acknowledgments

This research work was carried out as part of the collaborative research project on Artificial Intelligence (AI)-based Plant Phenomics in Plant Factories with Artificial Lighting (PFALs) financially supported by the New Energy and Industrial Technology Development Organization (NEDO). The author would like to thank all members in this project, especially: Hiroki Sawada and Tadashi Kudo of Kajima Corporation for capturing and providing image data; Prof. Toyoki Kozai, Eri Hayashi, Toshiji Ichinosawa and Dr. Yumiko Amagai of Japan Plant Factory Association for project management and discussion; Takeshi Nagami of AIST for discussion and comments. Finally, the author greatly appreciates the reviewers' effort for providing detailed and fruitful comments.

References

- [1] Ralf Metzner, Anja Eggert, Dagmar Dusschoten, Daniel Pflugfelder, Stefan Gerth, Uli Schurr, N Uhlmann, and Siegfried Jahnke. Direct comparison of MRI and X-ray CT technologies for 3D imaging of root systems in soil: Potential and challenges for root trait quantification. *Plant methods*, 11:17, 12 2015.
- [2] Miguel Piñeros, Brandon G Larson, Jon E Shaff, Dave Schneider, Alexandre Falcão, Lixing Yuan, Randy Clark, Eric Craft, Tyler Davis, Pierre-Luc Pradier, Nathanael M Shaw, Ithipong Assaranurak, Susan McCouch, Craig Sturrock, Malcolm Bennett, and Leon Kochian. Evolving technologies for growing, imaging, and analyzing 3D root system architecture of crop plants. *Journal of Integrative Plant Biology*, 58:230–241, 3 2016.
- [3] Rubén Rellán-Álvarez, Guillaume Lobet, Heike Lindner, Pierre-Luc Pradier, Jose Sebastian, Muh-Ching Yee, Yu Geng, Charlotte Trontin, Therese LaRue, Amanda Schrager Lavelle, Cara H. Haney, Rita Nieu, Julin Maloof, John P. Vogel, and José R. Dinneny. Glo-roots: an imaging platform enabling multidimensional characterization of soil-grown roots systems. *bioRxiv*, 2015. URL <https://www.biorxiv.org/content/early/2015/08/03/016931>.
- [4] Michael Adu, Antoine Chatot, Lea Wiesel, Malcolm J. Bennett, Martin Broadley, Philip White, and Lionel X. Dupuy. A scanner system for high-resolution quantification of variation in root growth dynamics of bras-

- sica rapa genotypes. *Journal of Experimental Botany*, 65, 2014.
- [5] Anand Seethepalli, Haichao Guo, Xiuwei Liu, Marcus Griffiths, Hussien Almtarfi, Zenglu Li, Shuyu Liu, Alina Zare, Felix Fritsch, Elison Blancaflor, Xuefeng Ma, and Larry M. York. Rhizovision crown: An integrated hardware and software platform for root crown phenotyping. *bioRxiv*, 2019. URL <https://www.biorxiv.org/content/early/2019/03/06/569707>.
- [6] Suxing Liu. 3D reconstruction of mazie roots system. URL <https://youtu.be/dmc8YZpmL2k>.
- [7] Suxing Liu and Alexander Bucksch. 3D root phenotyping for improved carbon sequestration. In *Phenome 2019*, 2019. URL https://www.researchgate.net/publication/331556832_3D_root_phenotyping_for_improved_carbon_sequestration.
- [8] Randy T. Clark, Robert B. MacCurdy, Janelle K. Jung, Jon E. Shaff, Susan R. McCouch, Daniel J. Aneshansley, and Leon V. Kochian. Three-dimensional root phenotyping with a novel imaging and software platform. *Plant Physiology*, June 2011.
- [9] Ying Zheng, Steve Gu, Herbert Edelsbrunner, Carlo Tomasi, and Philip Benfey. Detailed reconstruction of 3D plant root shape. In *Proceedings of the 2011 International Conference on Computer Vision*, pages 2026–2033, 2011.
- [10] Tomohiko Yano, Shohei Nobuhara, and Takashi Matsuyama. 3D shape from silhouettes in water for online novel-view synthesis. *IPSI Transactions on Computer Vision and Applications*, 5:65–69, July 2013.
- [11] Malte Pedersen, Stefan Hein Bengtson, Rikke Gade, Niels Madsen, and Thomas B. Moeslund. Camera calibration for underwater 3D reconstruction based on ray tracing using Snell’s law. In *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1491–14917, June 2018.
- [12] Sundera Tejaswi Digumarti, Gaurav Chaurasia, Aparna Taneja, Roland Siegwart, Amber Thomas, and Paul Beardsley. Underwater 3D capture using a low-cost commercial depth camera. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9, March 2016.
- [13] Anne Jordt, Kevin Kösera, and Reinhard Koch. Refractive 3D reconstruction on underwater images. *Methods in Oceanography*, 15:90–113, 2016.
- [14] Atsushi Yamashita, Etsukazu Hayashimoto, Toru Kaneko, and Yoshimasa Kawata. 3-D measurement of objects in a cylindrical glass water tank with a laser range finder. In *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*, volume 2, pages 1578–1583, 2003.
- [15] Kiriakos N. Kutulakos and Steven M. Seitz. A theory of shape by space carving. 38(3):199–218, 2000.
- [16] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM transactions on graphics (TOG)*, 23(3):309–314, 2004.
- [17] Yoshinobu Sato, Shin Nakajima, Nobuyuki Shiraga, Hideki Atsumi, Shigeyuki Yoshida, Thomas Koller, Guido Gerig, and Ron Kikinis. Three-dimensional multi-scale line filter for segmentation and visualization of curvilinear structures in medical images. *Medical Image Analysis*, 2(2):143–168, June 1988.
- [18] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, November 1986.
- [19] Agisoft. Metashape—photogrammetric processing of digital images and 3D spatial data generation. URL <https://www.agisoft.com/>.