Demeter: A Parametric Model of Crop Plant Morphology from the Real World

Tianhang Cheng, Akbert Zhai, Evan Z. Chen, Rui Zhou, Yawen Deng, Zitong Li
Kejie Zhao, Janice Shiu, Qianyu Zhao, Yide Xu, Xinlei Wang, Yuan Shen, Sheng Wang

Lisa Ainsworth, Kaiyu Guan, Shenlong Wang[†]
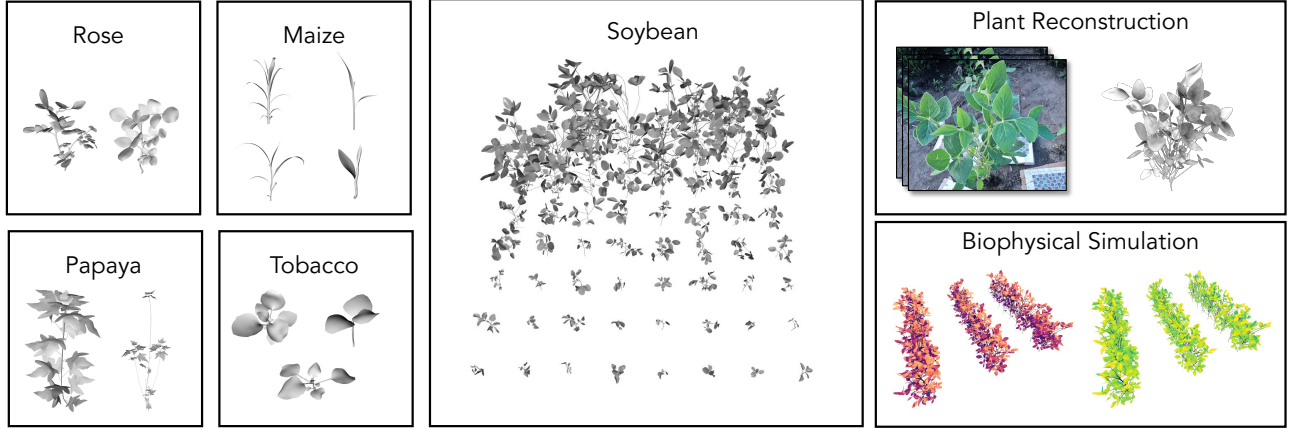University of Illinois Urbana-Champaign

Figure 1. **Demeter** is a parametric model that is trained on real-world data. It encodes plants into shape, topology, articulation, and deformations, providing a realistic, compact representation that can generalize to many species.

Abstract

Learning 3D parametric shape models of objects has gained popularity in vision and graphics and has showed broad utility in 3D reconstruction, generation, understanding, and simulation. While powerful models exist for humans and animals, equally expressive approaches for modeling plants are lacking. In this work, we present Demeter, a data-driven parametric model that encodes key factors of a plant morphology, including topology, shape, articulation, and deformation into a compact learned representation. Unlike previous parametric models, Demeter handles varying shape topology across various species and models three sources of shape variation: articulation, subcomponent shape variation, and non-rigid deformation. To advance crop plant modeling, we collected a large-scale, ground-truthed dataset from a soybean farm as a testbed. Experiments show that Demeter effectively synthesizes shapes, reconstructs structures, and simulates biophysical processes. Code and data is available at our project page.

1. Introduction

Crop plants are essential to life on Earth and human well-being. Food crops such as rice, maize, and soybeans form the foundation of global agriculture and feed billions of people; industrial crops supply crucial raw materials for manufacturing; tree crops produce oxygen, provide habitats, and help maintain ecological balance. Reconstructing, understanding, and modeling plants with computer vision opens pathways to enhance crop yields, monitor environmental health, and drive agricultural innovation. A critical step toward this goal is establishing a realistic, flexible, expressive, compact, and preferably data-driven morphological model for plants. While powerful parametric models [37, 45, 88] exist for human forms and quadrupedal animals, there is currently no equally expressive model for plants.

Our goal is to bridge the gap by building a compact parametric model capable of representing, reconstructing, animating, and simulating plants of a given species. To this end, we present **Demeter**, a novel, data-driven, category-based parametric model that encodes key factors driving variations in 3D plant morphology, including topology, articulation, shape variations, and non-articulated deformations (Fig. 2). The articulations of the plant body are modeled in a kine-
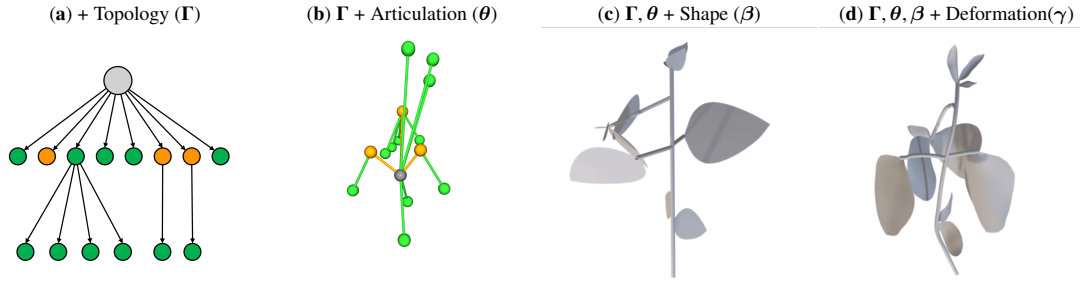
(a) + Topology (Γ)   **(b) Γ + Articulation (θ)**   **(c) Γ, θ + Shape (β)**   **(d) Γ, θ, β + Deformation(γ)**

Figure 2. The **Demeter Model** consists of four parametric components: (a) topology Γ, a tree-structured graph that stores the binary connection relationship between plant-part nodes; (b) articulation parameters θ defining each node's relative position to its parent, using quaternions for rotation and 1D translation; (c) shape parameters β, representing leaf variance via principal component coefficients, learned from 2D leaf scans; (d) deformation parameters γ, learned from 3D leaf point clouds and fixed 2D shape parameters β, controlling the 3D skeletons of leaf and stem. Here the gray, orange and green nodes represents root, stem and leaf respectively.

matic tree structure, with nodes representing elements of the plant and edges encoding joint angles that define the relative orientation between connected segments. We represent shapes and deformations of each leaf and stem using triangular meshes to ease reconstruction and simulation, where vertex locations are parameterized by Catmull-Rom spline curves. Although these shape and deformation parameters are high-dimensional, we observe—similar to findings in human and animal modeling—that their variations lie primarily in low-dimensional manifolds. Inspired by this, we fit and learn a compact linear space using principal components, significantly enhancing the model's compactness without compromising expressiveness.

The quality of a learned shape model depends on the scale and representativeness of our training data. To capture real-world variations, we collected **3D soybean** samples from a farm in Illinois over an entire growing season, capturing around 600 plants from multiple genotypes across all life stages—from germination to pod formation. Unlike lab environments, our row-crop farm samples minimize environmental discrepancies. We performed dense 3D reconstructions using a Gaussian splatting pipeline, manually corrected missing parts, and selected over 300 plants for detailed leaf and stem annotation. These annotated reconstructions trained the Demeter model for soybeans. Additionally, we used this dataset to develop benchmarks for 2D/3D semantic segmentation and 3D morphological shape reconstruction from images or point clouds.

We train Demeter for various species across different 2D and 3D datasets, including the aforementioned new soybean data. We rigorously evaluate the learned Demeter both qualitatively and quantitatively, testing its efficacy on tasks such as multi-view and monocular reconstruction, as well as downstream applications like photosynthesis and stomatal conductance simulations. Accurate plant morphology, as modeled by Demeter, enables detailed simulations for analyzing crop productivity, demonstrating its practical impact. Unlike conventional 3D models, Demeter provides a complete parametric morphological model, supporting diverse biophysical simulations and advancing scientific discovery.

To foster collaboration, we will open-source a Demeter-based platform, inviting vision and plant phenotyping researchers to create and share models of various species.

Our contributions are: 1) We introduce Demeter, a data-driven parametric model for plant morphology. 2) We demonstrate Demeter's capabilities in reconstruction, and simulation for multiple crop plant species. 3) We present a new dataset with over 300 soybeans, featuring 3D geometry, part segmentation, and topology annotations.

2. Related Works

Procedural Shape Procedural models aim to represent geometries by a series of programmatic operations to construct the geometry. They have been used extensively in computer graphics to generate building exteriors [33, 53, 54, 76] and interiors [16, 46, 49, 63], natural landscapes [8, 29, 52, 62], and entire cities [31, 56]. Most existing work on plant models is hand-crafted or procedural, incorporating biological knowledge to simulate the morphology of leaf [32, 47, 51, 72, 75] or the real growth of the plant [17, 34, 39, 59, 60, 85]. While these models yield realistic 3D outputs, their complex symbol-geometry relationship makes inverse modeling (e.g., estimating model parameters from images) challenging. In contrast, our model uses simple geometric transformations of parts (leaves and stems), making it easier to fit sensor observations.

Learning-based Parametric Shape To overcome the limitations of rule-based models and enhance expressiveness, researchers have used statistical learning to derive parametric models from real-world data, notably in the study of human body models [1, 2, 4, 13, 23, 45, 58, 81]. The most representative work is SMPL [45], which inspired our use of learned linear model. Unlike SMPL, which has a global dependency issue, Demeter learns separate, independent local shapes, offering greater flexibility to handle intra-species variations. The success of learned parametric models has also expanded to domains such as human faces [37], hands [65], quadrupedal animals [10, 15, 87, 88] and leaf [9]. Nevertheless, most current models focus on
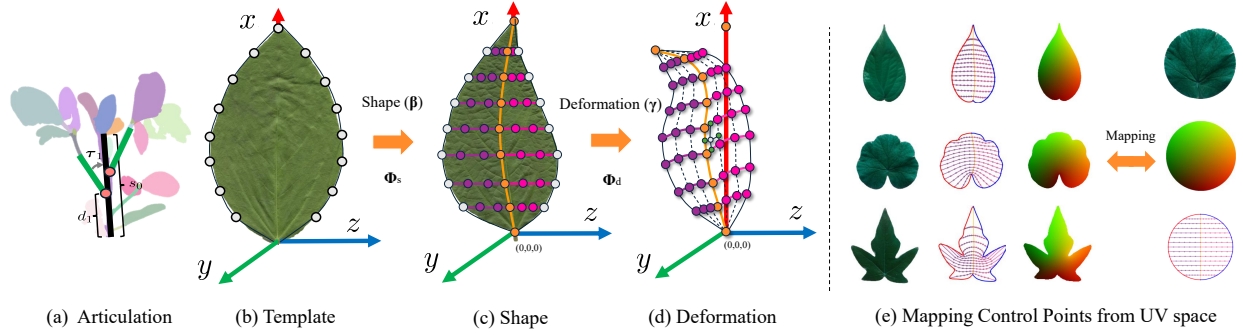
Figure 3. **Demeter model parameters**. (a) Articulation includes scale s, which controls the size of the component (e.g., stem length); position d, which controls the location of the joint node relative to its parent; and quaternion angle τ, which controls the relative rotation. (b) The mean template learned from 2D leaf scannings. (c) The shape parameter encodes the location of contour points, which defines the inner points of the leaf and serve as joints to deform the leaf and stem. (d) Deformation γ represents a grid of Catmull-Rom control points to manage shape and articulation-independent deformation, such as leaf curling. (e) An illustration of the mapping between the leaf shape control points and canonical UV space.

fixed topology and have not been applied to plant, which is still dominated by procedural or grammar-based models. Recent works [9, 12, 22, 28] starts to build on the parametric models of leaves but fail to model whole plants, or do not disentangle shape and deformation. Demeter aims to close this gap, allowing plant morphology to benefit from real-world data and statistical learning.

Inverse Parametric Shape Modeling Methods for reconstructing complete 3D plant shapes generally rely on inverse parametric modeling, especially inverse procedural modeling (IPM). IPM relies on fitting parameters of a procedural model based on structural similarity with the input observations. This approach has found success in fitting point cloud data [20, 21, 43, 71] and interactive user inputs [39, 44, 48]. To tackle reconstruction from single images, Li et al. [35] and SVDTree [38] train models (on procedurally generated data) to predict bounding volumes and then fill the volumes using space colonization [55]. These works focus on tree plants, and often ignore fine-grained leaf details. With Demeter, we aim to provide a plant morphology model that can accurately represent many kinds of variation in plant shapes while being conducive to inverse modeling due to its data-driven, primitive-based structure.

General 3D Reconstruction on Plants A number of works have investigated the use of generic reconstruction methods to model plant shapes. The methods used include multi-view stereo [73, 74, 78, 79, 86] and neural radiance fields [5, 25, 66, 70]. However, without incorporating shape priors, such methods struggle outside clean lab settings due to plants' thin structures and heavy occlusion. Demeter incorporates prior knowledge and data-driven models to constrain the solution space for 3D reconstruction.

3. Method

Our parametric model, Demeter, is illustrated in Fig. 2. Like parametric models for humans and animals [45, 88], it de-

composes the mesh into shape vectors and deformation vectors. However, Demeter additionally encodes the variable structural topology of plant components. We fit species-specific Demeter models from real-world 3D scans and learn linear bases of shape and deformation. In the following sections, we detail the formulation and parameterization of Demeter (Sec. 3.1), describe the procedure for learning Demeter-Soybean from real-world data (Sec. 3.2), and finally demonstrate how to use learned Demeter models for image-based and point-based 3D reconstruction (Sec. 3.3).

3.1. Demeter Plant Model

Demeter is a model that represents each plant's shape with a set of parameters that uniquely depict the plant's vegetative morphology.

$$(\mathbf{V}, \mathbf{F}) = \mathcal{M}(\mathbf{\Gamma}, \boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{\gamma}), \tag{1}$$

where $(\mathbf{V}, \mathbf{F})$ are the output vertices and faces, respectively; $\mathbf{\Gamma}, \boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{\gamma}$ represent topology, articulation, shape, and non-articulated deformation, respectively. In essence, Demeter is a function $\mathcal{M} : \mathbb{Z}_2^{|\mathbf{\Gamma}|} \times \mathbb{R}^{|\boldsymbol{\theta}| \times |\boldsymbol{\beta}| \times |\boldsymbol{\gamma}|} \to \mathbb{R}^{3 \times |\mathbf{V}|} \times \mathbb{Z}^{3 \times |\mathbf{F}|}$ that maps a set of parameters to a triangulated plant mesh. Demeter also a learn a compact representation $\mathbf{\Phi} = \{\mathbf{\Phi}_s, \mathbf{\Phi}_d\}$ about shape $\boldsymbol{\beta}$ and deformation $\boldsymbol{\gamma}$ for each species. Specifically, given a Demeter model $\mathcal{M}$, we calculate vertices of the plant using the following equation:

$$\mathbf{v}_f = \mathrm{T}(\boldsymbol{\theta}; \mathbf{\Gamma}) \cdot \mathcal{D}(\mathbf{v}_t + \mathcal{S}(\boldsymbol{\beta}); \boldsymbol{\gamma}) \tag{2}$$

where $\mathbf{v}_t$ is the vertices location of the templates, learned from the mean shape of each organ; $\mathbf{v_f} \in \mathbf{V}$ is the final location of the deformed plant; $\mathcal{S}(\boldsymbol{\beta})$ represents the shape (grid points) offset of each part that $\mathbf{v}_t$ resides in; $\mathcal{D}(\mathbf{v}; \boldsymbol{\gamma})$ denotes a shape-independent deformation; T is composed of rigid + scale transformations along the kinematic chain defined by $\mathbf{\Gamma}$, parameterized by the articulation $\boldsymbol{\theta}$. Next, we describe each element of Demeter in detail.

Topology Unlike in human or animal models, instances within the same plant species can have significantly different topology. We represent the plant topology $\mathbf{\Gamma}$ as a tree data structure of $n = n_l + n_s + n_o$ nodes, each representing an individual part of the plant, e.g. a leaf or stem. We currently omit other structures (e.g. flowers and fruit) for simplicity. Each node i has a type $\text{tp}(i) \in \{\text{leaf}, \text{stem}\}$ and is connected to its parent $\text{pa}(i) \in \mathbb{Z}$ (except the root).

Articulation The articulation is represented as a set of per-node joint state parameters $\boldsymbol{\theta} = \{\theta_i\}_{i=1}^n$, where each $\boldsymbol{\theta_i} = (\tau_i, d_i, s_i)$ represents the rotation quaternion $\tau_i \in \mathbb{R}^4$, path length $d_i \in [0, 1]$ and scale $s_i \in \mathbb{R}^+$ of its canonical space relative to the local coordinate system of its parent stem $\text{pa}(i)$, as shown in Fig. 3 (a). The root node serves as the origin. For vertex $\mathbf{v}$ belonging to i-th component, a forward kinematic chain will be computed, in which the affine transform $\mathbf{T}$ of each part is computed through affine composition from the part to the root:

$$\mathbf{T}_i(\boldsymbol{\theta}; \mathbf{\Gamma}) = \prod_{i' \in \text{ans}(i)} \mathrm{T}(\tau'_\mathrm{i}, \mathrm{d}'_\mathrm{i}, \mathrm{s}'_\mathrm{i}) \qquad (3)$$

where $\mathbf{T}(\tau'_i, d'_i, s'_i)$ is an affine transform from the parent $\text{pa}(i)$ to current node i and $\text{ans} = \{i, \text{pa}(i), \text{pa}(\text{pa}(i)), ...\}$ denotes the ordered set of ancestor nodes of i (the kinematic chain). Details see Fig. 3 (a).

Shape The shape parameters $\boldsymbol{\beta} = \{\beta_i\}_{i=1}^{n_l+n_s}$ define the offsets to the template shape in the canonical space for each organ instance, assuming no 3D deformations. This is conceptually similar to the T-pose for humans, depicting the intrinsic identity-preserving shape factor. For *stems*, the template is parameterized as a unit length cylinder with m_s uniform control points (also as joints here) on the axis. And the shape parameter $\boldsymbol{\beta}$ only controls its thickness. For *leaves*, the template is a $m_{l_1} \times m_{l_2}$ 2D grid enclosed by a left and right contour learned from the mean of the 2D leaf scannings. The grid points of each horizontal grid line serve as the control points of a Catmull-Rom curve [11], whose control points always lie on the curve, making it easier to interpret and manipulate than other parametric curves, such as Bezier curves [18]. To further reduce dimensionality, we compress the parameters by learning a linear model $\mathbf{\Phi}_s \in \mathbb{R}^{2m_{l_1}m_{l_2} \times |\beta|}$, such that $\mathbf{\Phi}_s^T \boldsymbol{\beta}$ represents the offset for two contour curves, where $|\boldsymbol{\beta}| \ll 2m_{l_1}m_{l_2}$. We learn the linear model $\mathbf{\Phi}_s$ by running PCA over thousands of 2D scans of leaf (Sec. 3.2). This provides a formal definition of shape offset:

$$\mathbf{v}_t + \mathcal{S}(\boldsymbol{\beta}) = \mathbf{\Phi}_s^T \boldsymbol{\beta} + \mathbf{v}_t, \qquad (4)$$

For non-grid surface points, we use the analytical Catmull-Rom spline interpolation based on control points. We refer readers to the supplement for more details. Fig. 4 illustrates the learned shape model for leaf of several species.

Deformation Shape and articulation define internal factors of vertex geometry, while external forces like wind, gravity, and contact cause deformations. Modeling these deformations is crucial for biophysical processes (e.g., photosynthesis, growth). Directly controlling vertex locations is inefficient and unrealistic, as real-world deformations typically obey constraints like length and area preservation.

To this end, we turn the joints of leaf into a 2D skeletal structure to drive the 3D deformation. This gives us a fixed topological structure, with one "main vein" in the middle and multiple horizontal "sub-veins". These veins comprised a chain structure as shown in Fig. 3 (c). Moreover, these joints along with the contour points naturally form a quadmesh for each leaf, which we can easily subdivide and triangulate. For stems, we directly connect all joints into a 1D skeleton.

We could use the joint state along the skeletal structure to encode deformation. For the j-th control point on the 2D skeleton, a joint angle ψ_j indicates its rotation relative to its parent node. We use forward kinematics to convert the angles into deformed 3D locations:

$$\mathcal{D}(\mathbf{v}_j; \gamma_i) = \prod_{j' \in \text{ans}(j)} \mathrm{T}(\tau_{\mathrm{j}'}, \mathrm{d}_\mathrm{j}, 1) \cdot \mathbf{v}_\mathrm{j}, \qquad (5)$$

where $\mathbf{v}_j$ is the input 3D vertex location before deformation in node i, $\text{ans}(j)$ denotes the ancestors of control point j, $\mathbf{T}$ is the rigid transformation corresponding to each joint angle, and d_j is the shape-defined skeletal distance between control point j and its parent d_j. Such deformation enjoys many benefits: 1) strong local rigidity preservation, making leaf area and stem length stay roughly unchanged; 2) ease of calculation, 3) ease of compression due to the spatial smoothness of the joint angles. Fig. 3 (c) depicts the deformation process.

Each leaf has 1 main vein, and each main control points has 2 sub-veins. The deformation parameter count is $2m_{l_1}m_{l_2}$ since each control point has 2 degrees of freedom. Inspired by the success of FLAME [36] in compressing expression-related deformations, we again adopt a linear dimensionality reduction model to represent deformation as $\mathbf{\Phi}_d^T \gamma$, where $\mathbf{\Phi}_d \in \mathbb{R}^{2m_{l_1}m_{l_2} \times |\gamma|}$ such that $\mathbf{\Phi}_d^T \gamma$ approximates the entire deformation field with $|\gamma| \ll 2m_{l_2}$. We learn $\mathbf{\Phi}_s$ by running PCA over real-world data, later described in Sec. 3.2.

3.2. Learning Demeter from Real-World Data

Demeter model parameters $\mathbf{\Phi}$ can be learned using real-world plant data captured in the crop field, grounding the learned Demeter model with realistic in-the-wild variations. To demonstrate this, we collect multi-view field data to recover complete geometry, scan 2D leaf images, employ annotators to label each plant component, and finally acquire a clean topology $\mathbf{\Gamma}$ (see Fig. 8). The shape parameter basis $\mathbf{\Phi}_s$ and joints $\mathbf{\Phi}_j$ is then learned by fitting contours from
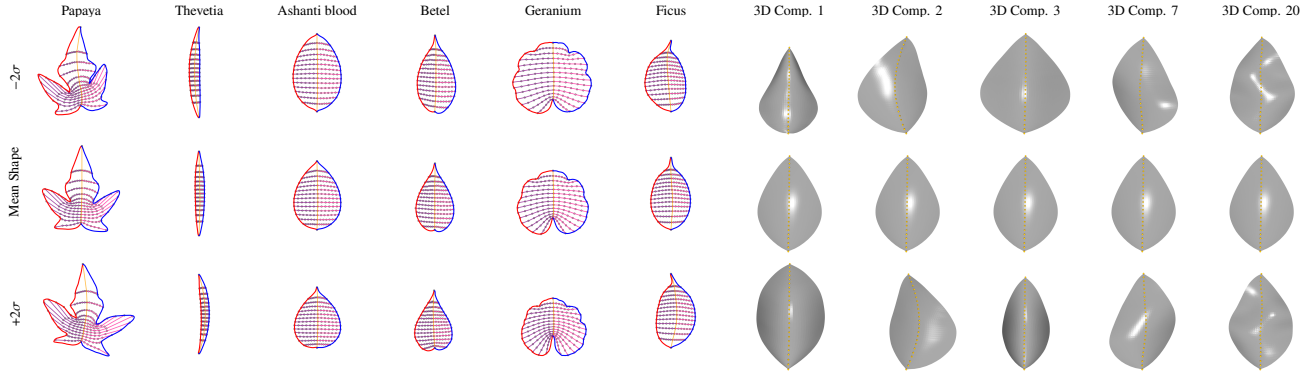
Figure 4. **PCA coefficients of leaf shape.** We visualize the mean shape and the one of the principal components of leaf for several species in $(-2\sigma, 0, 2\sigma)$. For 3D deformation, we show the first and last few components for soybean.
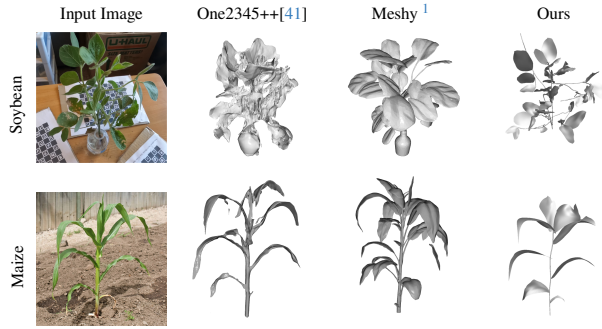

Figure 5. **Reconstruction from single image**. One2345++ generates a coarse shape but fails to capture details. Meshy reconstructs a realistic plant but is misaligned with the input image. In contrast, our model is complete and faithful to input.
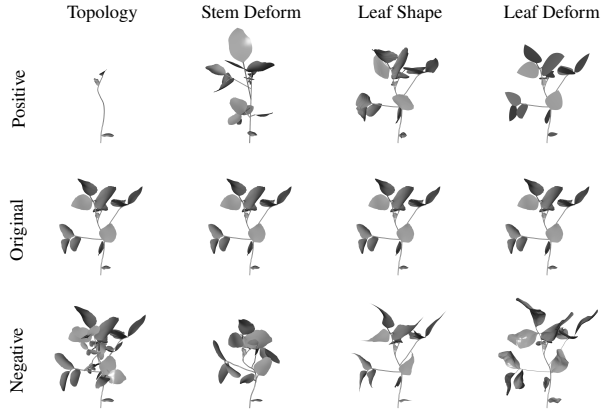

Figure 6. **Parameter Interpolation.** We could explicitly change the overall shape and topology while keeping the identity. We apply the same changes for all nodes for better visualization, but in practice we could manipulate each instance individually.

2D leaf scans using PCA. The deformation parameter basis $\mathbf{\Phi}_d$ is derived by fitting real-world 3D deformation data from annotated scans, followed by structured joint optimization. We refer to our model as Demeter-Soybean. Note that this entire process has little species-specific design and can therefore generalize to other plant species.

Data Collection and Processing   Most 3D plant scans for phenotyping are conducted in controlled lab environments, which may lack the realism and variability of real-world field conditions. To address this issue, we created a custom annotated 3D soybean dataset with 600 soybean samples of various growth stages from a smart soybean farm. Using these samples, we captured over 600 multi-view RGB videos with a GoPro in both indoor and outdoor settings.

For 3D geometry extraction, we first applied structure-from-motion with Hloc [67, 68] to determine camera poses. We then used 4DGS [77] to reconstruct dynamic plants affected by wind. Although 4DGS captures high-quality renderings and motion, it lacks precision in surface geometry. To address this, we set the $t = 0$ and freeze movement to render a set of static multi-view images. We then apply 2DGS [26] on these images to extract a high-quality mesh.

The reconstructed mesh sometimes had missing parts, particularly stems, which we manually corrected. We also annotated instance segmentation and topology to provide ground-truth data, as shown in Fig. 8. These annotations are essential for training the deformation model and will serve as benchmarks for 3D reconstruction.

Learning Shape Basis from 2D Data   We derive the shape parameter basis, $\mathbf{\Phi}_s$ from real-world 2D leaf scans in the FGLIR [3] dataset and Folio dataset [61]. All images are calibrated to have accurate metric scale.

To extract the contours, we first annotate two key points on each leaf—the *base* (where the leaf connects to the stalk) and the *tip*, and optionally some keypoints in the middle. These annotations enable us to align each leaf along the x-axis. Second, we extract the leaf mask using SAM2 [64] and extract the entire contour from the mask. Third, we divide the contour into two parts according to the keypoints.

Next, we construct the inner points of a leaf by dividing it into a grid with $m_{l_1} \times m_{l_2}$ control points given the contours. For a *simply connected* leaf with general shapes, we first numerically map the control points from canonical UV space to each 2D leaf scanning, and then learn the linear model from these data. Inspired by [69], we model this problem
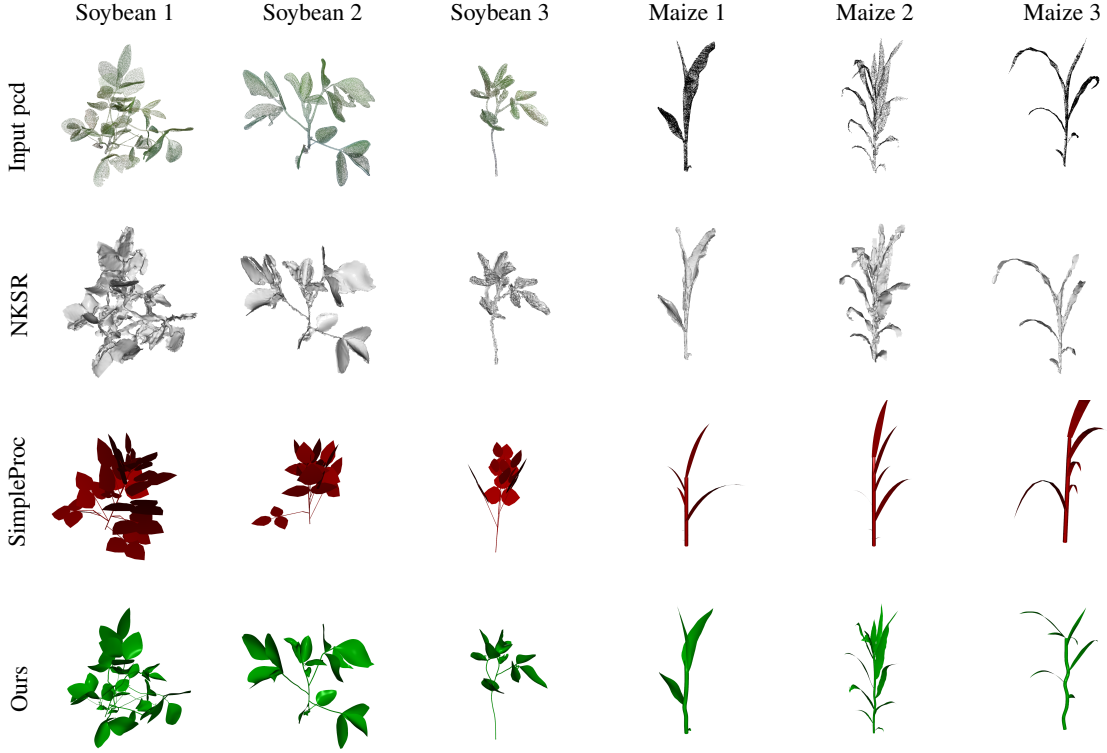
Figure 7. **Reconstruction from 3D point cloud.** The results show that our model fits the input more accurately than the baselines.

as a unique mapping $\varphi : \Omega \rightarrow \Omega'$ between UV domain Ω' and leaf domain Ω. Specifically, we define the boundary mapping $b : \partial\Omega \rightarrow \partial\Omega'$ by mapping the left/right contour of the leaf to the left/right contour of a circle, and solve Laplace equation

$$\Delta\varphi = 0, \text{ s.t. } \varphi|_{\partial\Omega} = b, \qquad (6)$$

to get the mapping φ and its inverse φ^{-1} numerically, forming a smooth bijective mapping (Fig. 3(e)). Particularly, for some oval leaves, such as soybean, whose contours have at most two intersections with the horizontal lines, we could simply build this mapping by linearly interpolating between each pair of intersections, and the positions of these control points are also linearly dependent on the shape parameter β. Fig. 3 (c, e) illustrates how a leaf shape is controlled by such contour curves.

Finally, we perform PCA on these control points, resulting in a linear model $\mathbf{\Phi}_s \in \mathbb{R}^{2m_{l_1}m_{l_2} \times |\beta|}$ capturing the variation in shape. The PCA components of soybean are visualized in Fig. 4.

Learning Deformation Basis from 3D Data Unlike leaf and stem shapes, which can be observed directly from 2D scans, extracting deformation is challenging due to entanglement with articulation and shape factors. Thus, we design a scheme to disentangle these factors through joint optimization, enabling us to learn the deformation basis $\mathbf{\Phi}_d$. Formally, given the input 3D point cloud collection, $\mathcal{P}_{i...N}$, where N is the number of input point cloud, ground-truth annotated topology, $\mathbf{\Gamma}_{i...N}$, semantic and instance labels,

$\mathbf{tp}_{i...N}$, and pretrained shape basis, $\mathbf{\Phi}_s$, our goal is to estimate the articulation parameters, θ_i, and shape parameters, $\beta_{i...N}$. Additionally, we aim to estimate $\gamma_{i...N}$, the dense, articulation- and shape-independent deformation for each plant, which will be used to train the PCA model $\mathbf{\Phi}_d$. We formulate this as an energy minimization problem:

$$\min_{\gamma, \beta, \theta} \text{CD}(\mathcal{P}, \{T(\theta; \mathbf{\Gamma}) \cdot \mathcal{D}(\mathbf{v}_t + \mathcal{S}(\beta); \gamma)\}_{\forall \mathbf{v}_t}) \qquad (7)$$

where CD is Chamfer distance, $\mathbf{v}_t$ represents canonical template vertex, $\mathbf{\Gamma}$ is input topology, $\mathcal{S}(\beta)$ is the shape model in Eq. 4 and $\mathcal{D}(\mathbf{v}; \beta)$ is the deformation function in Eq. 5 and $T(\theta, \mathbf{\Gamma})$ is the kinematic articulation in Eq. 3.

Joint inference is challenging due to nonlinearity and complex forward kinematics. Inspired by prior work [42, 82], we adopt a multi-stage approach. For each segment, we first fit a coarse shape parameter, $\hat{\beta}$, and estimate its rigid pose in world coordinates without articulation constraints (e.g., leaves detached from stems). Using this pose, we perform forward kinematic fitting to incorporate articulation, obtaining an initial model, $\hat{\theta}$. Deformation is initialized to identity, i.e., $\mathcal{D}(\cdot; \hat{\beta})$. With these initializations, we jointly optimize the energy in Eq. 7, resulting in our final estimation as shown in the right of Fig. 8.

Finally, we learn the linear PCA basis $\mathbf{\Phi}_d$ from the real-world deformations $\beta_{1...N}$ to compress them into a compact space without loss of expressiveness, shown in Fig.4.
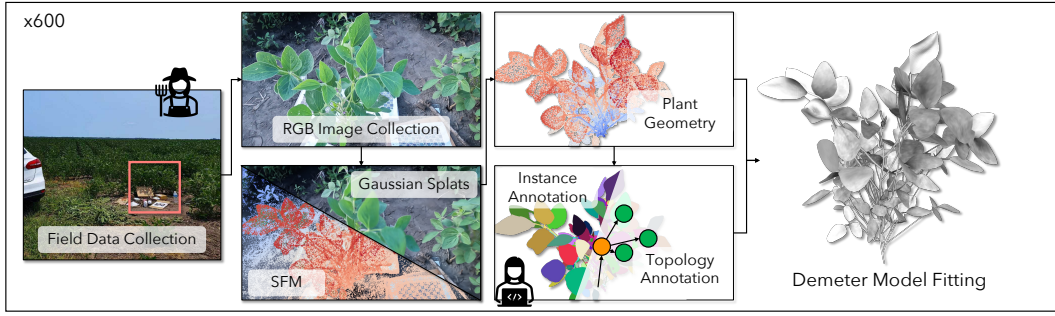
Figure 8. **Demeter-soybean data collection process**. We collect around 600 monocular RGB video of soybean in the field, and run a Gaussian Splatting based multiview reconstruction pipeline to get the mesh. After that, we manually annotate the instance segmentation and the topology, and fit this ground-truth data with our Demeter parametric model. A similar process can be adapted to other species.

3.3. 3D Plant Reconstruction with Demeter

Demeter encodes strong morphological priors about the plant. Inspired by the success of SMPL for human reconstruction [19, 30], we investigate the use of Demeter parameters for multi-view and single-view reconstruction.

Multiview Reconstruction Using multiview images, we first recover the point cloud with 2DGS, like in Demeter training. However, unlike before, there is no ground-truth instance mask or topology annotation available during real-world inference. Hence, we adopt a three-step strategy. First, we perform instance segmentation on the recovered 3D point cloud using a PointTransformer-based network. Next, we infer the topology based on a minimum spanning tree. Finally, we fit parameters as in Eq. 7.

We trained a PointTransformer-V3 [80] based instance segmentation network. Inspired by prior work in 2D [6], we predict semantic logits for each point and its truncated inverse distance to the instance boundary. Using the inferred (thresholded) instance boundaries, we remove boundary points to isolate connected components. We then run DBSCAN with the same threshold to cluster points into individual leaves and stems. We find this approach effective and visualizations can be found in the supplementary.

Given the instance predictions, the topology Γ is automatically inferred by finding a minimal spanning tree rooted from the main stem. Finally, we recover the parametric model using the described method for Eq. 7.

Single Image Reconstruction We present a baseline method for single-view plant reconstruction to showcase the usefulness of Demeter for constraining 3D geometry. Given a single RGB plant image, we first apply the pretrained SAM [64] to get the mask, and use Mask-RCNN [24] on the masked image to predict instance segmentation with 3 classes: leaf, stem, and main stem.

We apply an off-the-shelf depth estimator[83] to lift all the instances to 3D, and perform simple filtering (details in supplementary). Then we infer the topology from the partial point cloud via minimal spanning tree, filtering stems without any children. Afterwards, we fit the Demeter model

	NKSR [27]	SimpProc [84]	Ours
Smooth		√	√
Disentangle		√	√
Learnable	√		√
Soybean, CD ↓	0.0030	0.0376	0.0016
Maize, CD ↓	0.0023	0.0557	0.0071
Soybean, size (KB) ↓	5785.6	0.2754	3.3750
Maize, size (KB) ↓	3686.4	0.2236	1.7656

Table 1. **Quantitative 3D reconstruction results.** We report the storage size in kilobytes (KB) and Chamfer distance (CD) in normalized scale. Results show that we fit the input point cloud better than the procedural method [84]. NKSR outputs large storage size because it stores dense mesh. We also achieve comparable CD while using much less parameters. We highlight the best and second best values.

parameters in the same way as before, but constraining the shape β and adopt L1 Chamfer distance for robustness.

4. Experiments

We evaluate the Demeter model across several aspects: fitting (Sec. 4.1), reconstruction (Sec. 4.2) and interpretability (Sec. 4.3). Additional training details and quantitative results are in the supp. material.

4.1. Fitting

Demeter can represent plants of different species, capturing individual leaves and stems with high fidelity (Fig. 9). We select Pepper and Rose from the PLANesT3D [50] dataset, and Tobacco from the PLANT3D [14] dataset. In Fig. 7, we compare Demeter with SimpleProc, a baseline procedural model adapted from CropCraft [84]. SimpleProc has a compact set of 8 global shape parameters such as average leaf length and node count, and is fitted by Bayesian optimization of the Chamfer distance with the ground truth point cloud. The results show that the Demeter-Soybean is much more expressive and can capture the individual leaf and stem shapes that SimpleProc cannot.
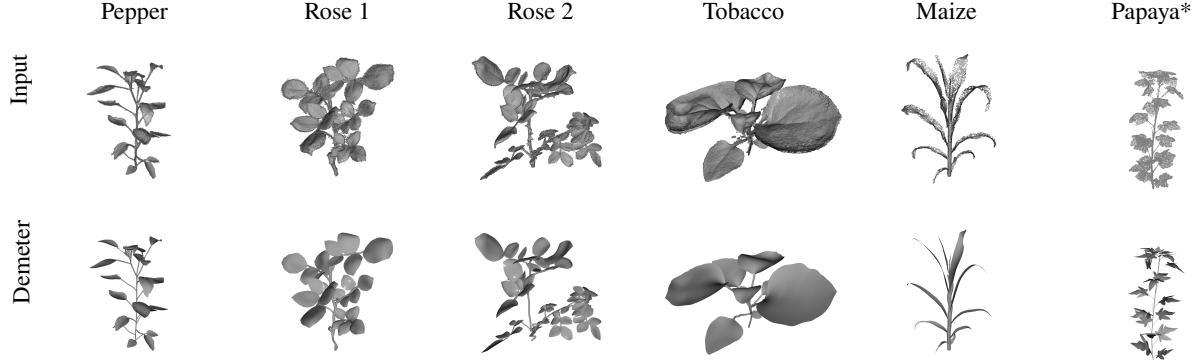
4.2. Reconstruction

Figure 9. **Generalization**. We learn Demeter on each species (pepper, rose, tobacco, maize) and validate on novel test shapes. Here we use the Papaya model to fit the point cloud of Ribes since we lack 2D leaf scans for Ribes and these two species have similar leaves.

Multi-view Reconstruction We train PointTransformer-V3 on 80 cleaned soybean point clouds and 5 maize point clouds for the segmentation module and evaluate the reconstruction pipeline on unseen testing samples. We compare our approach against SimpleProc [84], an inverse procedural generation framework based on L-system, and NKSR [27], the state-of-the-art neural SDF-based 3D reconstruction pipeline. As shown in Fig. 7 and Tab. 1, our model performs well on unseen samples, recovering realistic geometry with high fidelity to the input. SimpleProc is compact but misaligned with the input, while NKSR [27] achieves low error yet produces a triangular mesh with thick, unsmooth leaves and stems and poor connectivity, preventing a biophysically functional plant model.

Single-view Reconstruction We trained Mask-RCNN on 32K images across 60 plants. The qualitative results of the single view reconstruction are shown in Fig. 5. We further evaluate the IOU between the rendered mask and the input as a quantaitive alignment score. From the figure, we see that Demeter achieves better alignment score (IoU: 0.328) with the input than Meshy (IoU: 0.206) and Zero12345++ [40] (IoU: 0.296). In contrast, our baselines produce less accurate shapes, struggling with the complexity of plant structures.

4.3. Discussions

Latent Space Visualization The Demeter model allows explicit control of deformation, shape, and topology while preserving plausibility (Fig. 6). For topology, we cut branches to create smaller trees or duplicate subtrees to enlarge them. Leaf and shape deformations are adjusted using PCA coefficients, and stem deformation is linearly manipulated from straight to curved.

Biophysical Simulation Applications Demeter provides high-quality, realistic 3D meshes of plants that can be fed directly into simulation software to predict important agroecosystem variables. We showcase this capability by gener-
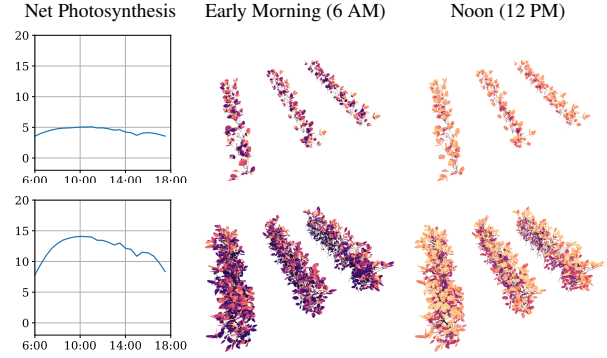

Figure 10. **Photosynthesis simulation results**. We perform simulations using Helios [7] on two soybean canopies generated by repeating Demeter-Soybean models. Left: timeseries of the net photosynthesis rate for the crop canopy over the course of a day, in units of $\mu molCO_2/m^2/s$. Other columns: mesh visualization where each leaf face is colored according to the rate of photosynthesis over that face (brighter = higher rate).

ating small crop fields by placing fitted Demeter models in a grid, and passing it to Helios [7] to simulate the response of the plant to weather variations over the course of a day. The weather variables were taken from data measured by a flux tower [57] and include temperature, humidity, radiation, and other environmental variables. In Fig. 10, we visualize the output of the photosynthesis rate.

Limitations We assume leaves are 2D shapes and stems are curves with uniform thickness. This excludes species with complex structures (e.g., cacti, algae, banyan trees) but applies to many common species.

5. Conclusion

We presented Demeter, a parametric shape model for plants using knowledge-based primitives and learning. Using a real-world soybean dataset, we show that our Demeter-Soybean model faithfully represents plant shapes and supports reconstruction and simulation applications.

6. Acknowledgment

References

[1] Brett Allen, Brian Curless, and Zoran Popović. The space of human body shapes: reconstruction and parameterization from range scans. *ACM transactions on graphics (TOG)*, 22 (3):587–594, 2003. 2

[2] Brett Allen, Brian Curless, Zoran Popović, and Aaron Hertzmann. Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 147–156. Citeseer, 2006. 2

[3] Ryuhei Ando, Yuko Ozasa, and Wei Guo. Robust surface reconstruction of plant leaves from 3d point clouds. *Plant Phenomics*, 2021. 5

[4] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: shape completion and animation of people. In *ACM SIGGRAPH 2005 Papers*, pages 408–416. 2005. 2

[5] Muhammad Arbab Arshad, Talukder Jubery, James Afful, Anushrut Jignasu, Aditya Balu, Baskar Ganapathysubramanian, Soumik Sarkar, and Adarsh Krishnamurthy. Evaluating nerfs for 3d plant geometry reconstruction in field conditions. *arXiv preprint arXiv:2402.10344*, 2024. 3

[6] Min Bai and Raquel Urtasun. Deep watershed transform for instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5221–5229, 2017. 7

[7] Brian N Bailey. Helios: A scalable 3d plant and environmental biophysical modeling framework. *Frontiers in Plant Science*, 10:1185, 2019. 8

[8] Fares Belhadj and Pierre Audibert. Modeling landscapes with ridges and rivers: bottom up approach. In *Proceedings of the 3rd international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, pages 447–450, 2005. 2

[9] Derek Bradley, Derek Nowrouzezahrai, and Paul Beardsley. Image-based reconstruction and synthesis of dense foliage. *ACM Transactions on Graphics (TOG)*, 32(4):1–10, 2013. 2, 3

[10] Thomas J Cashman and Andrew W Fitzgibbon. What shape are dolphins? building 3d morphable models from 2d images. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):232–244, 2012. 2

[11] E Catmull. A class of local interpolating splines. *Computer Aided Geometric Design/Academic Press*, 1974. 4

[12] Gaurav Chaurasia and Paul Beardsley. Editable parametric dense foliage from 3d capture. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5305–5314, 2017. 3

[13] Yinpeng Chen, Zicheng Liu, and Zhengyou Zhang. Tensor-based human body modeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 105–112, 2013. 2

[14] Adam Conn, Ullas V Pedmale, Joanne Chory, and Saket Navlakha. High-resolution laser scanning reveals plant architectures that reflect universal network design principles. *Cell systems*, 5(1):53–62, 2017. 7

[15] Jake Deane, Sinead Kearney, Kwang In Kim, and Darren Cosker. Dynadog+ t: A parametric animal model for synthetic canine image generation. *arXiv preprint arXiv:2107.07330*, 2021. 2

[16] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, and Roozbeh Mottaghi. Procthor: Large-scale embodied ai using procedural generation. *Advances in Neural Information Processing Systems*, 35:5982–5994, 2022. 2

[17] Oliver Deussen, Pat Hanrahan, Bernd Lintermann, Radomír Měch, Matt Pharr, and Przemyslaw Prusinkiewicz. Realistic modeling and rendering of plant ecosystems. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 275–286, 1998. 2

[18] Gerald Farin, Josef Hoschek, and M-S Kim. *Handbook of computer aided geometric design*. Elsevier, 2002. 4

[19] Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa, and Jitendra Malik. Humans in 4d: Reconstructing and tracking humans with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14783–14794, 2023. 7

[20] Jianwei Guo, Shibiao Xu, Dong-Ming Yan, Zhanglin Cheng, Marc Jaeger, and Xiaopeng Zhang. Realistic procedural plant modeling from multiple view images. *IEEE Transactions on Visualization and Computer Graphics*, 26:1372–1384, 2020. URL https://api.semanticscholar.org/CorpusID:52284731. 3

[21] Jan Hackenberg, Heinrich Spiecker, Kim Calders, Mathias I. Disney, and Pasi Raumonen. Simpletree —an efficient open source tool to build tree models from tls clouds. *Forests*, 6:4245–4294, 2015. URL https://api.semanticscholar.org/CorpusID:4236665. 3

[22] Mozhgan Hadadi, Mehdi Saraeian, Jackson Godbersen, Talukder Jubery, Yawei Li, Lakshmi Attigala, Aditya Balu, Soumik Sarkar, Patrick S Schnable, Adarsh Krishnamurthy, et al. Procedural generation of 3d maize plant architecture from lidar data. *arXiv preprint arXiv:2501.13963*, 2025. 3

[23] Nils Hasler, Carsten Stoll, Martin Sunkel, Bodo Rosenhahn, and H-P Seidel. A statistical model of human pose and body shape. In *Computer graphics forum*, volume 28, pages 337–346. Wiley Online Library, 2009. 2

[24] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 7

[25] Kewei Hu, Ying Wei, Yaoqiang Pan, Hanwen Kang, and Chao Chen. High-fidelity 3d reconstruction of plants using neural radiance field. *arXiv preprint arXiv:2311.04154*, 2023. 3

[26] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 5

[27] Jiahui Huang, Zan Gojcic, Matan Atzmon, Or Litany, Sanja Fidler, and Francis Williams. Neural kernel surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4369–4379, 2023. 7, 8

[28] SoHyeon Jeong, Si-Hyung Park, and Chang-Hun Kim. Simulation of morphology changes in drying leaves. In *Computer Graphics Forum*, volume 32, pages 204–215. Wiley Online Library, 2013. 3

[29] K Raiyan Kamal and Yusuf Sarwar Uddin. Parametrically controlled terrain generation. In *Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia*, pages 17–23, 2007. 2

[30] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7122–7131, 2018. 7

[31] George Kelly and Hugh McCabe. Citygen: An interactive system for procedural city generation. In *Fifth International Conference on Game Design and Technology*, pages 8–16. Liverpool, 2007. 2

[32] Daeyeoul Kim and Jinmo Kim. Procedural modeling and visualization of multiple leaves. *Multimedia Systems*, 23: 435–449, 2017. 2

[33] Mathieu Larive and Veronique Gaildrat. Wall grammar for building generation. In *Proceedings of the 4th international conference on Computer graphics and interactive techniques in Australasia and Southeast Asia*, pages 429–437, 2006. 2

[34] Jae Joong Lee, Bosheng Li, and Bedrich Benes. Latent l-systems: Transformer-based tree generator. *ACM Transactions on Graphics*, 43(1):1–16, 2023. 2

[35] Bosheng Li, Jacek Ka lużny, Jonathan Klein, Dominik L Michels, Wojtek Pa lubicki, Bedrich Benes, and Sören Pirk. Learning to reconstruct botanical trees from single images. *ACM Transactions on Graphics (TOG)*, 40(6):1–15, 2021. 3

[36] Tianye Li, Timo Bolkart, Michael. J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6):194:1–194:17, 2017. URL https://doi.org/10.1145/3130800.3130813. 4

[37] Tianye Li, Timo Bolkart, Michael J Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4d scans. *ACM Trans. Graph.*, 36(6):194–1, 2017. 1, 2

[38] Yuan Li, Zhihao Liu, Bedrich Benes, Xiaopeng Zhang, and Jianwei Guo. Svdtree: Semantic voxel diffusion for single image tree reconstruction. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4692–4702, 2024. URL https://api.semanticscholar.org/CorpusID:271122336. 3

[39] Bernd Lintermann and Oliver Deussen. Interactive modeling of plants. *IEEE Computer Graphics and Applications*, 19(1): 56–65, 1999. 2, 3

[40] Minghua Liu, Ruoxi Shi, Linghao Chen, Zhuoyang Zhang, Chao Xu, Xinyue Wei, Hansheng Chen, Chong Zeng, Jiayuan Gu, and Hao Su. One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. *arXiv preprint arXiv:2311.07885*, 2023. 8

[41] Minghua Liu, Ruoxi Shi, Linghao Chen, Zhuoyang Zhang, Chao Xu, Xinyue Wei, Hansheng Chen, Chong Zeng, Jiayuan Gu, and Hao Su. One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10072–10083, 2024. 5

[42] Shaowei Liu, Saurabh Gupta, and Shenlong Wang. Building rearticulable models for arbitrary 3d objects from 4d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21138–21147, 2023. 6

[43] Yotam Livny, Feilong Yan, Matt Olson, Baoquan Chen, Hao Zhang, and Jihad El-Sana. Automatic reconstruction of tree skeletal structures from point clouds. In *ACM SIGGRAPH Asia 2010 papers*, pages 1–8. 2010. 3

[44] Steven Longay, Adam Runions, Frédéric Boudon, and Przemyslaw Prusinkiewicz. Treesketch: Interactive procedural modeling of trees on a tablet. In *SBIM@ Expressive*, pages 107–120. Citeseer, 2012. 3

[45] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 851–866. 2023. 1, 2, 3

[46] Ricardo Lopes, Tim Tutenel, Ruben M Smelik, Klaas Jan De Kraker, and Rafael Bidarra. A constrained growth method for procedural floor plan generation. In *Proc. 11th Int. Conf. Intell. Games Simul*, pages 13–20. Citeseer, 2010. 2

[47] Shenglian Lu, Chunjiang Zhao, and Xinyu Guo. Venation skeleton-based modeling plant leaf wilting. *International Journal of Computer Games Technology*, 2009(1):890917, 2009. 2

[48] Gilda Manfredi, Nicola Felice Capece, Ugo Erra, and Monica Gruosso. Treesketchnet: From sketch to 3d tree parameters generation. *ACM Transactions on Intelligent Systems and Technology*, 14:1 – 29, 2022. URL https://api.semanticscholar.org/CorpusID:251041067. 3

[49] Fernando Marson and Soraia Raupp Musse. Automatic real-time generation of floor plans based on squarified treemaps algorithm. *International Journal of Computer Games Technology*, 2010(1):624817, 2010. 2

[50] Kerem Mertoğlu, Yusuf Şalk, Server Karahan Sarıkaya, Kaya Turgut, Yasemin Evrenesoğlu, Hakan Çevikalp, Ömer Nezih Gerek, Helin Dutağacı, and David Rousseau. Planest-3d: A new annotated dataset for segmentation of 3d plant point clouds. *arXiv preprint arXiv:2407.21150*, 2024. 7

[51] Teng Miao, ChunJiang Zhao, XinYu Guo, and ShengLian Lu. A framework for plant leaf modeling and shading. *Mathematical and Computer Modelling*, 58(3-4):710–718, 2013. 2

[52] Gavin SP Miller. The definition and rendering of terrain maps. In *Proceedings of the 13th annual conference on Com-

puter graphics and interactive techniques, pages 39–48, 1986. 2

[53] Pascal Müller, Peter Wonka, Simon Haegler, Andreas Ulmer, and Luc Van Gool. Procedural modeling of buildings. In *ACM SIGGRAPH 2006 Papers*, pages 614–623. 2006. 2

[54] Pascal Müller, Gang Zeng, Peter Wonka, and Luc Van Gool. Image-based procedural modeling of facades. *ACM Trans. Graph.*, 26(3):85, 2007. 2

[55] Wojciech Palubicki, Kipp Horel, Steven Longay, Adam Runions, Brendan Lane, Radomír Měch, and Przemyslaw Prusinkiewicz. Self-organizing tree models for image synthesis. *ACM Transactions On Graphics (TOG)*, 28(3):1–10, 2009. 3

[56] Yoav IH Parish and Pascal Müller. Procedural modeling of cities. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 301–308, 2001. 2

[57] Gilberto Pastorello, Carlo Trotta, Eleonora Canfora, Housen Chu, Danielle Christianson, You-Wei Cheah, Cristina Poindexter, Jiquan Chen, Abdelrahman Elbashandy, Marty Humphrey, et al. The fluxnet2015 dataset and the oneflux processing pipeline for eddy covariance data. *Scientific data*, 7(1):225, 2020. 8

[58] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10975–10985, 2019. 2

[59] Przemyslaw Prusinkiewicz and Aristid Lindenmayer. *The algorithmic beauty of plants*. Springer Science & Business Media, 2012. 2

[60] Przemyslaw Prusinkiewicz, Jim Hanan, and Radomír Měch. An L-system-based plant modeling language. In *International workshop on applications of graph transformations with industrial relevance*, pages 395–410. Springer, 1999. 2

[61] S. Pudaruth. Folio. UCI Machine Learning Repository, 2015. DOI: https://doi.org/10.24432/C5TW3F. 5

[62] Alexander Raistrick, Lahav Lipson, Zeyu Ma, Lingjie Mei, Mingzhe Wang, Yiming Zuo, Karhan Kayan, Hongyu Wen, Beining Han, Yihan Wang, et al. Infinite photorealistic worlds using procedural generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12630–12641, 2023. 2

[63] Alexander Raistrick, Lingjie Mei, Karhan Kayan, David Yan, Yiming Zuo, Beining Han, Hongyu Wen, Meenal Parakh, Stamatis Alexandropoulos, Lahav Lipson, et al. Infinigen indoors: Photorealistic indoor scenes using procedural generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21783–21794, 2024. 2

[64] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 5, 7

[65] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *arXiv preprint arXiv:2201.02610*, 2022. 2

[66] Farah Saeed, Jin Sun, Peggy Ozias-Akins, Ye Juliet Chu, and Changying Charlie Li. Peanutnerf: 3d radiance field for peanuts. In *CVPRW*, pages 6253–6262, 2023. 3

[67] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *CVPR*, 2019. 5

[68] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning feature matching with graph neural networks. In *CVPR*, 2020. 5

[69] Teseo Schneider and Kai Hormann. Smooth bijective maps between arbitrary planar polygons. *Computer Aided Geometric Design*, 35:243–254, 2015. 5

[70] Claus Smitt, Michael Halstead, Patrick Zimmer, Thomas Läbe, Esra Guclu, Cyrill Stachniss, and Chris McCool. Pagnerf: Towards fast and efficient end-to-end panoptic 3d representations for agricultural robotics. *IEEE Robotics and Automation Letters*, 9(1):907–914, 2023. 3

[71] Ondrej Stava, Sören Pirk, Julian Kratt, Baoquan Chen, Radomír Měch, Oliver Deussen, and Bedrich Benes. Inverse procedural modelling of trees. In *Computer Graphics Forum*, volume 33, pages 118–131. Wiley Online Library, 2014. 3

[72] Xiaopeng Sun, Jia Fu, Teng Chen, and Yu Dong. Wrinkle and curl distortion of leaves using plant dynamic. *Graphical Models*, 118:101118, 2021. 2

[73] Yongzhe Sun, Zhixin Zhang, Kai Sun, Shuai Li, Jianglin Yu, Linxiao Miao, Zhanguo Zhang, Yang Li, Hongjie Zhao, Zhenbang Hu, et al. Soybean-mvs: annotated three-dimensional model dataset of whole growth period soybeans for 3d plant organ segmentation. *Agriculture*, 13(7):1321, 2023. 3

[74] Feiyi Wang, Xiaodan Ma, Meng Liu, and Bingxue Wei. Three-dimensional reconstruction of soybean canopy based on multivision technology for calculation of phenotypic traits. *Agronomy*, 12(3):692, 2022. 3

[75] Hanxiao Wang, Biao Zhang, Jonathan Klein, Dominik L Michels, Dongming Yan, and Peter Wonka. Autoregressive generation of static and growing trees. *arXiv preprint arXiv:2502.04762*, 2025. 2

[76] Peter Wonka, Michael Wimmer, François Sillion, and William Ribarsky. Instant architecture. *ACM Transactions on Graphics (TOG)*, 22(3):669–677, 2003. 2

[77] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20310–20320, 2024. 5

[78] Sheng Wu, Weiliang Wen, Yongjian Wang, Jiangchuan Fan, Chuanyu Wang, Wenbo Gou, and Xinyu Guo. Mvs-pheno: a portable and low-cost phenotyping platform for maize shoots using multiview stereo 3d reconstruction. *Plant Phenomics*, 2020. 3

[79] Sheng Wu, Weiliang Wen, Wenbo Gou, Xianju Lu, Wenqi Zhang, Chenxi Zheng, Zhi Yang Xiang, Liping Chen, and

Xinyu Guo. A miniaturized phenotyping platform for individual plants using multi-view stereo 3d reconstruction. *Frontiers in Plant Science*, 13, 2022. URL https://api.semanticscholar.org/CorpusID:251372887. 3

[80] Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler faster stronger. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4840–4851, 2024. 7

[81] Hongyi Xu, Eduard Gabriel Bazavan, Andrei Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Ghum & ghuml: Generative 3d human shape and articulated pose models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6184–6193, 2020. 2

[82] Gengshan Yang, Minh Vo, Natalia Neverova, Deva Ramanan, Andrea Vedaldi, and Hanbyul Joo. Banmo: Building animatable 3d neural models from many casual videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2863–2873, 2022. 6

[83] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv preprint arXiv:2406.09414*, 2024. 7

[84] Albert J. Zhai, Xinlei Wang, Kaiyuan Li, Zhao Jiang, Junxiong Zhou, Sheng Wang, Zhenong Jin, Kaiyu Guan, and Shenlong Wang. Cropcraft: Inverse procedural modeling for 3d reconstruction of agricultural crops. *arXiv preprint arXiv:2411.09693*, 2024. 7, 8

[85] Xiaochen Zhou, Bosheng Li, Bedrich Benes, Songlin Fei, and Sören Pirk. Deeptree: Modeling trees with situated latents. *IEEE Transactions on Visualization and Computer Graphics*, 2023. 2

[86] Rongsheng Zhu, Kai Sun, Zhuangzhuang Yan, Xuehui Yan, Jianglin Yu, Jia Shi, Zhenbang Hu, Hongwei Jiang, Dawei Xin, Zhanguo Zhang, et al. Analysing the phenotype development of soybean plants using low-cost 3d reconstruction. *Scientific Reports*, 10(1):7055, 2020. 3

[87] Silvia Zuffi, Angjoo Kanazawa, David W Jacobs, and Michael J Black. 3d menagerie: Modeling the 3d shape and pose of animals. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6365–6373, 2017. 2

[88] Silvia Zuffi, Ylva Mellbin, Ci Li, Markus Hoeschle, Hedvig Kjellström, Senya Polikovsky, Elin Hernlund, and Michael J Black. Varen: Very accurate and realistic equine network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5374–5383, 2024. 1, 2, 3

Demeter: A Parametric Model of Plant Morphology from the Real World

Supplementary Material
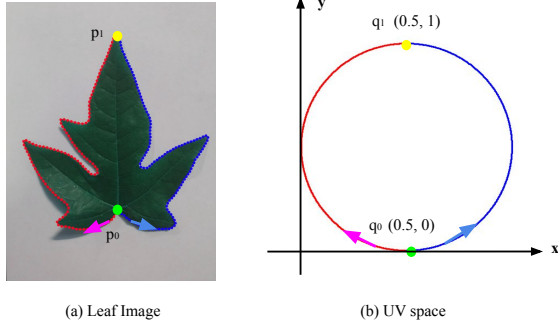

(a) Leaf Image   (b) UV space

Figure 3. **Initial value creation for Laplace equation.** We map the boundary value from the template to the leaf according to the curve length along the contour.

A. Catmull-Rom Curve

Given a shape template with the skeleton and deformation, we generate a smooth geometry with Catmull-Rom curve. A general Catmull-Rom curve is composed of $K + 2$ control points $\{p\}_{i=0}^{K+2}$ and $K + 1$ segments. The first and the last control point p_0, p_{K+1} are virtual control points, which depend on other control points, i.e. $p_0 = p_1 - (p_2 - p_1)$, $p_{K+1} = p_K + (p_K - p_{K-1})$, as shown in the blue dots in Fig. 2.

Each segment $\mathbf{p}(t)$ in time $t \in [0, 1]$ is a cubic polynomial curve and defined by the consecutive four control points: $\mathbf{p}_{i-2}$, $\mathbf{p}_{i-1}$, $\mathbf{p}_i$, and $\mathbf{p}_{i+1}$, and a shape parameter α. We set $\alpha = 0.5$ for all segments in this paper. Each segment satisfies $\mathbf{p}(0) = \mathbf{p}_{i-1}$ and $\mathbf{p}(1) = \mathbf{p}_i$. Assuming $\mathbf{p}(t) = [\mathbf{p}_{i-2}, \mathbf{p}_{i-1}, \mathbf{p}_i, \mathbf{p}_{i+1}] \mathbf{C}$, the blending weights $\mathbf{C}(t)$ are given by:

$$\mathbf{C}(t) = \begin{bmatrix} -\alpha t + 2\alpha t^2 - \alpha t^3 \\ 1 + (\alpha - 3)t^2 + (2 - \alpha)t^3 \\ \alpha t + (3 - 2\alpha)t^2 + (\alpha - 2)t^3 \\ -\alpha t^2 + \alpha t^3 \end{bmatrix}. \quad (1)$$

B. Detailed explaination of reconstruction pipeline

Fig. 1 depicts our single- or multi-image inference pipeline.

3D Geometry We first acquire a point cloud—either by estimating camera intrinsics and depth with Perspective-Field and DepthAnything on a single view and lifting to a partial cloud, or by running SfM+GSplats on multi-view images to obtain a full cloud.

Parameter	Meaning
i	the index of node
j	the index of control points within a node
$\mathcal{M}$	the Demeter Model
$\mathbf{F}$	the faces set of Demeter mesh
$\mathbf{V}$	the vertices of Demeter mesh
v_f	the vertices of the templates after shape offset and deformation, $v_f \in \mathbf{V}$
$\mathbf{\Phi}$	the parameter of Demeter model (PCA coefficient)
$\mathbf{\Gamma}$	topology
pa(i)	the parent of the node i
ans(j)	the parent of the control point j (within node i)
$\boldsymbol{\theta}$	articulation
$\boldsymbol{\theta}_i$	the articulation of node i
$\boldsymbol{\tau}_i$	rotation of node i
d_i	length with respect to the parent stem of node i
s_i	scale of node i
T	rigid + scale transformations
$\mathbf{T}$	accumulated rigid + scale transformations along the kinematic chain
$\boldsymbol{\beta}$	shape
$\boldsymbol{\beta}_i$	the shape parameter of node i
v_t	the vertices of the templates without deformation (canonical space)
$\mathcal{S}$	shape operation to template vertices
$\mathbf{\Phi}_s$	PCA of Shape
$\boldsymbol{\gamma}$	deformation
d_j	length with respect to parent of control point j
$\boldsymbol{\tau}_j$	rotation of control point j
v_j	vertex j in node i with shape offset
$\mathcal{D}$	deformation operation to template vertices
$\mathbf{\Phi}_d$	PCA of Deformation
tp(i)	the type of the node i
n	total number of nodes
n_l	the number of leaf node
n_s	the number of stem node
n_o	the number of other node except leaf and stem
m_{l_1}	the number of vertical control points of a leaf
m_{l_2}	the number of horizontal control points of a leaf
m_s	the number of control points of a stem
Ω	UV space
Ω'	leaf space
φ	the mapping between UV space and leaf
∂	the boundary of domain
b	the mapping between the boundary of UV space and leaf
$\mathcal{P}$	the input point cloud
N	the number of input points

Table 1. Explanation of Demeter model notation.

Segmentation We then perform instance segmentation—using MaskRCNN on the single view and unprojecting masks to 3D, or using PointTransformer-v3 (PT-v3) on the multi-view cloud to predict per-point categories and inverse-distance scores, removing high-score points and clustering with DBSCAN.
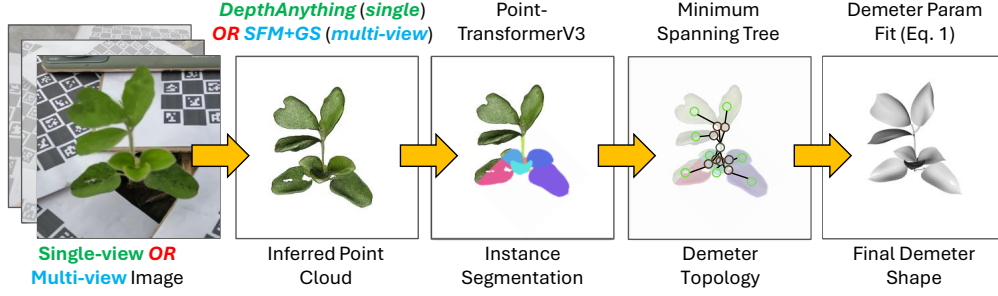
Figure 1. **Test-time Reconstruction Pipeline.** The pipeline transforms the input images to a segmented point cloud, extracts the topology and fits part templates, and finally outputs the parametric mesh.
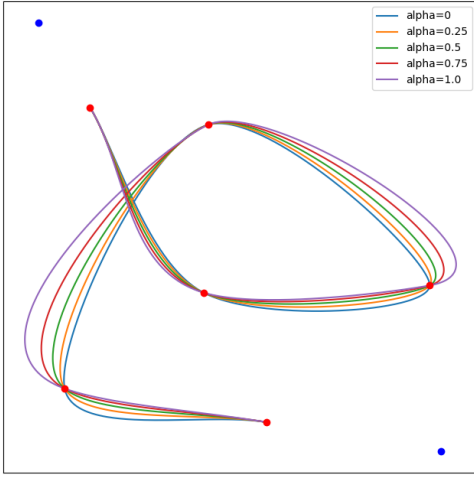

Figure 2. **Visualization of Catmull-Rom Curve.** The alpha parameter controls the curvature of the curve. We choose $\alpha = 0.5$ in this paper. The red dots represent the control points and the blue ones represent the virtual control points, which is necessary to draw the first and last segment.

Tree-structure recovery We construct a dense directed graph G over the estimated instances. The edge weights W_{ij} enforce the constraint that leaves have no children: $W_{ij} = 0$ if $i = j$; $W_{ij} = \infty$ if cluster i is leaf; otherwise $W_{ij} = r_{ij}$, where r_{ij} is the min distance of any points from cluster i to j. Then, we compute the minimum spanning tree (MST) over G starting from the root stem, which yields the plant skeleton with the minimum connection cost.

Demeter Fitting Finally, given the estimated topology, seg, and point cloud, we fit each cluster with a leaf or stem template, then run a global optimization over all parts parameters by minimizing Chamfer distance between the parametric shape and input cloud, producing the final Demeter parametric mesh.

C. Additional Model Details

Topology Not every arbitrary tree structure corresponds to a valid plant. Most plants exhibit strong morphological constraints. For example, each plant may have a maximum depth or width and exhibit some first-order constraints, including the following: leaves cannot possess child nodes, and for soybeans, top-canopy stems often have a triplet of leaves as children. We enforce the former constraint by pruning edges with leaves as parents before constructing the minimal spanning tree. Incorporation of further constraints is left for future work.

Articulation Each node, except the root, is connected to its parent stem. The articulation of each node determines the rigid transformation relative to the local coordinate system at the connection point, as shown in Fig. 9. Given an arbitrary stem, the coordinate system depends only on the stem shape and the position of the connection point, and is independent of the global 6 DoF pose of the stem.

Given m_s control points $\{v\}_{j=1}^{m_s}$ on the stem, we calculate the relative rotation $\{R_j\}_{j=1}^{m_s-1}$ of control points from each segment $v_{j+1} - v_j$, where $j = 1, 2, ..., m_s - 1$, and the y-axis of R_0 is aligned with the first segment. The points in each segment will have the same local coordinate as their preceding control point.

Shape We solve Eq. 6 in the main paper to get the bijective mapping between the leaf and template. To get the initial boundary value, we assign the leaf bottom p_0 to $q_0 = (0.5, 0)$, and the leaf tip p_1 to $q_1 = (0.5, 1.0)$. For other points along the contour, we uniformly map the boundary value according to the curve length of the point, as shown in Fig. 3.

D. Additional Results and Visualizations

Overfitting over different species To further show the generalization ability, we report the overfitting result on

Figure 4. **Render Demeter in Blender**. We could attach textures (Fig. 5) to the leaf of our Demeter model and dump to rendering engine such as blender for photo-realistic rendering. From left to right, the image shows pepper, soybean and maize respectively.
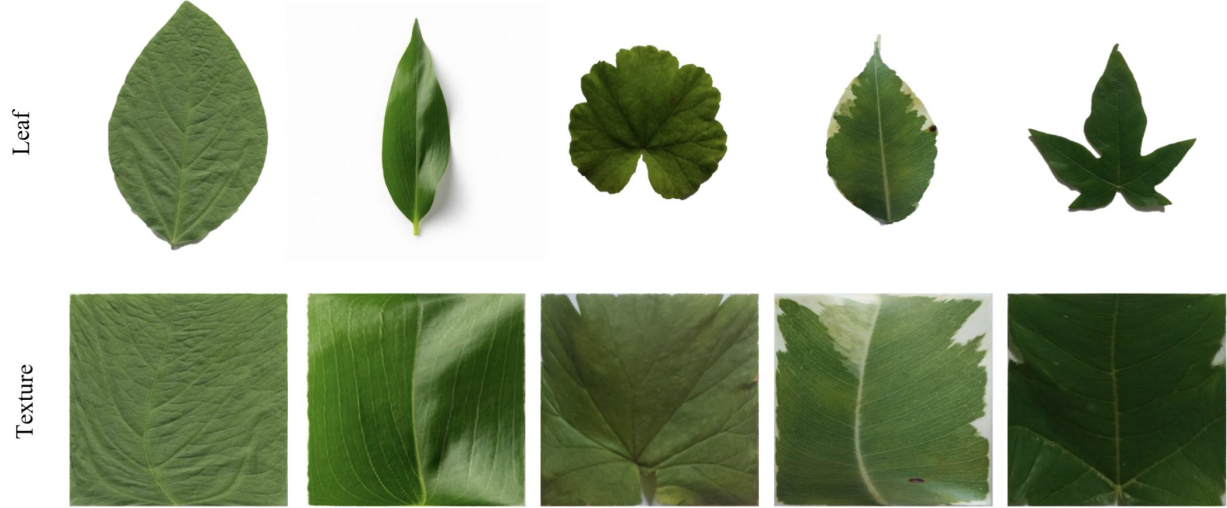

Figure 5. **The texture extracted from leaf images using bijective mapping.** Similarly, we could map the leaf to a square UV space and get the texture, which can be easily applied to the grid points of our parametric leaf.

	Ribes	Rose	Pepper	Tobacco	Soybean	Maize	Smooth	Learnable	Disentangle
NKSR	0.084	0.095	0.087	0.081	0.174	0.332		✓	
Bezier	0.134	0.066	0.082	0.252	0.159	0.978	✓		
Ours	0.121	0.041	0.054	0.095	0.154	0.649	✓	✓	✓

Table 2. **Leaf fitting error for different species.** We show normalized CD($\times$ 100). We highlight the best and second best values.

different species using our model, NKSR and Bézier surfaces (most common NURBS shape) and report results in

Tab. 2. We didn't compare with CropCraft[6] since it only works for Soybean. Our model yields superior reconstruc-
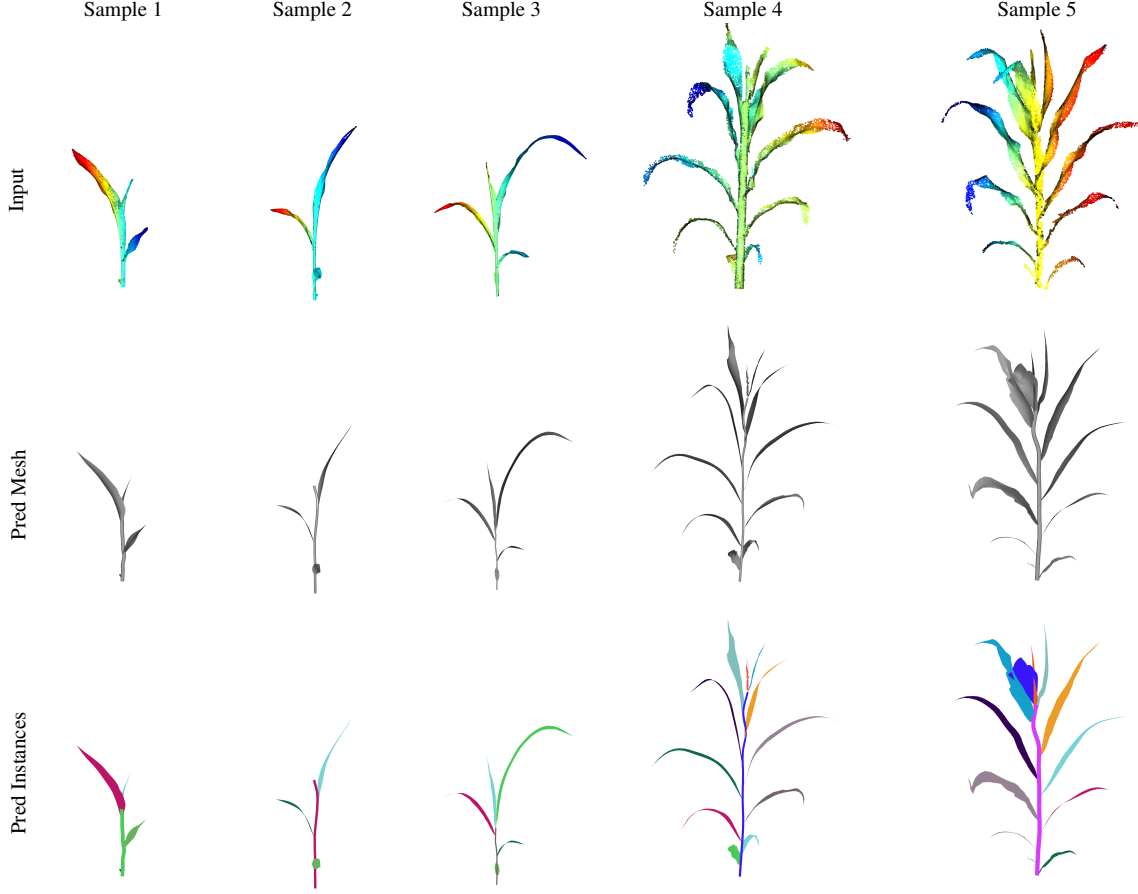
Figure 6. **Qualitative results of reconstructing maize.** We show the results of fitting Demeter-Maize models to point clouds from the Pheno4D dataset (sample 1,2 and 3) and web dataset [7] (sample 4 and sample 5). Different colors represent different instances. The results show that our model can generalize to species beyond soybeans and capture shape details.
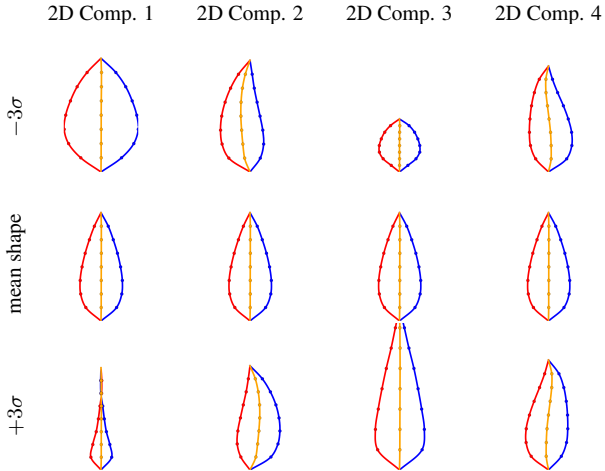

Figure 7. **PCA coefficient of soybean leaf.** We visualize the first several principle component of leaf for soybean in $(-3\sigma, 3\sigma)$.

tion plus two key benefits: (1) a PCA-based, learnable shape prior; and (2) disentangled, interpretable parame-ters—unlike Bézier curves, whose control points lack phys-ical meaning and are not even on-surface, our on-leaf and on-vein control points and PCA basis correspond to bio-physical/phenotypical traits.

3D Reconstruction Given the ground-truth instance seg-mentation, we calculate the minimal distance from each point to other instances, as shown in Fig. 10.

For training, we follow the default configuration in PointTransformer-V3 but without the mixing strategy. We set the batch size as 7 and use AdamW optimizer with a learning rate 0.0025, weight decay 0.02, and trained in 300 epochs. The loss function is composed of the cross-entropy loss and distance regression loss, i.e. $l_{total} = l_{ce} + 15l_{dist}$.

The predicted graph differs from the ground truth due to both errors in prediction and the inherent ambiguity in stem connectivity labeling. Additionally, our model cannot automatically fill in missing stems if the input point cloud has too many missing parts, as shown in Fig. 13.
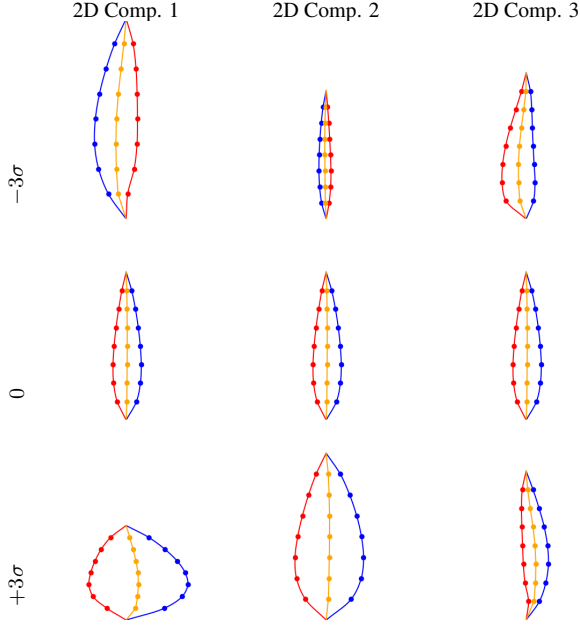
Figure 8. **PCA coefficients of Demeter-Maize leaf shape.** We vary the PCA coefficients for maize leaf shape across $(-3\sigma, 3\sigma)$ with respect to the mean shape for all 3 components. The red and blue points represent the control points for the 2D contour and the orange curve represents the initial skeleton for 3D deformation.
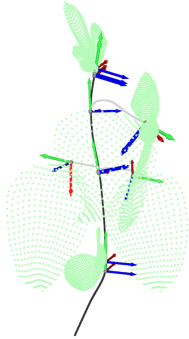

Figure 9. **Local coordinate system on the stem.** Each point on the curve is defined in a local coordinate system determined by the curve's intrinsic geometry at that specific point.

2D Reconstruction We train Mask-RCNN with a ResNet50-FPN backbone starting from the COCO-pretrained model in detectron2 [4] for 20K iterations. We use the default configuration but with a batch size of 4 and lowering the learning rate at 8K and 12K iterations. For inference, we use a confidence threshold of 0.8.

We apply an off-the-shelf depth estimator [5] to lift all the instances to 3D. However, the lifted partial 3D point clouds are usually noisy. Therefore we remove instances if the total number of points is less than 30 for stems and 100 for leaves, and remove leaves based on the proportions of
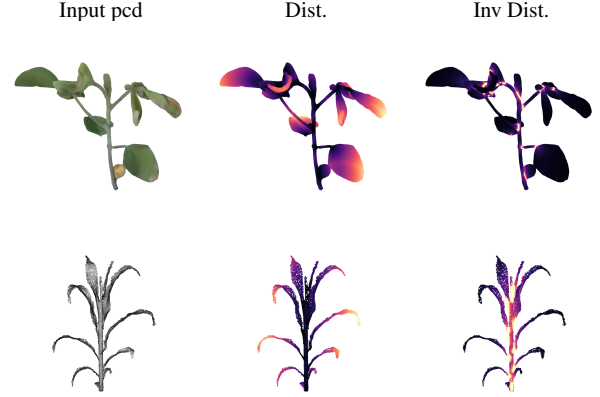

Figure 10. **Truncated Inverse Distance.** Given a input point cloud with instance segmentation, we select the point in every instance and calculate the distance to all other instances for that point (Middle). Afterwards, we select a threshold and calculate the inverse distance and train PointTransformer to predict this value.

their rotated bounding boxes. We also filter out the points with a high gradient of depth. We infer the topology from the partial point cloud by building a minimal spanning tree, and removing the "bare stems" (stems without any children) to reduce noise. Afterwards, we fit the Demeter model parameters in the same way as before. During fitting, we apply the linear model to constrain the shape parameter β into $(-2\sigma, 2\sigma)$ and adapt the L1 chamfer distance for robustness to outliers. As a result, our model achieves a 2D IoU of 0.9028 between the mask of predicted mesh and the mask of ground-truth mesh.

Our model may produce noisy outputs in novel views because we only apply constraints at the node level, not at the global topology level. We also did not constrain the leaf deformation or stem deformation. We leave these as directions for future improvements.

Remark Both the multi-view and single-image reconstruction we proposed here is a preliminary exploration of 3D reconstruction using Demeter. Despite showing great potential, we believe there is significant potential for the community to develop better reconstruction algorithms in the future using our parametric model.

E. Detail about Maize Species

We show that the Demeter methodology generalizes to other plant species by developing a **Demeter-Maize** prototype using the Pheno4D [3] dataset, which consists of 84 maize point clouds capturing different growth stages of 6 maize plants. We also reconstruct maize from a web dataset [7], which contains larger maize plants compared to Pheno4D. To acquire the 2D shape parameterization, we simply append an additional scaling axis to the soybean leaf shape
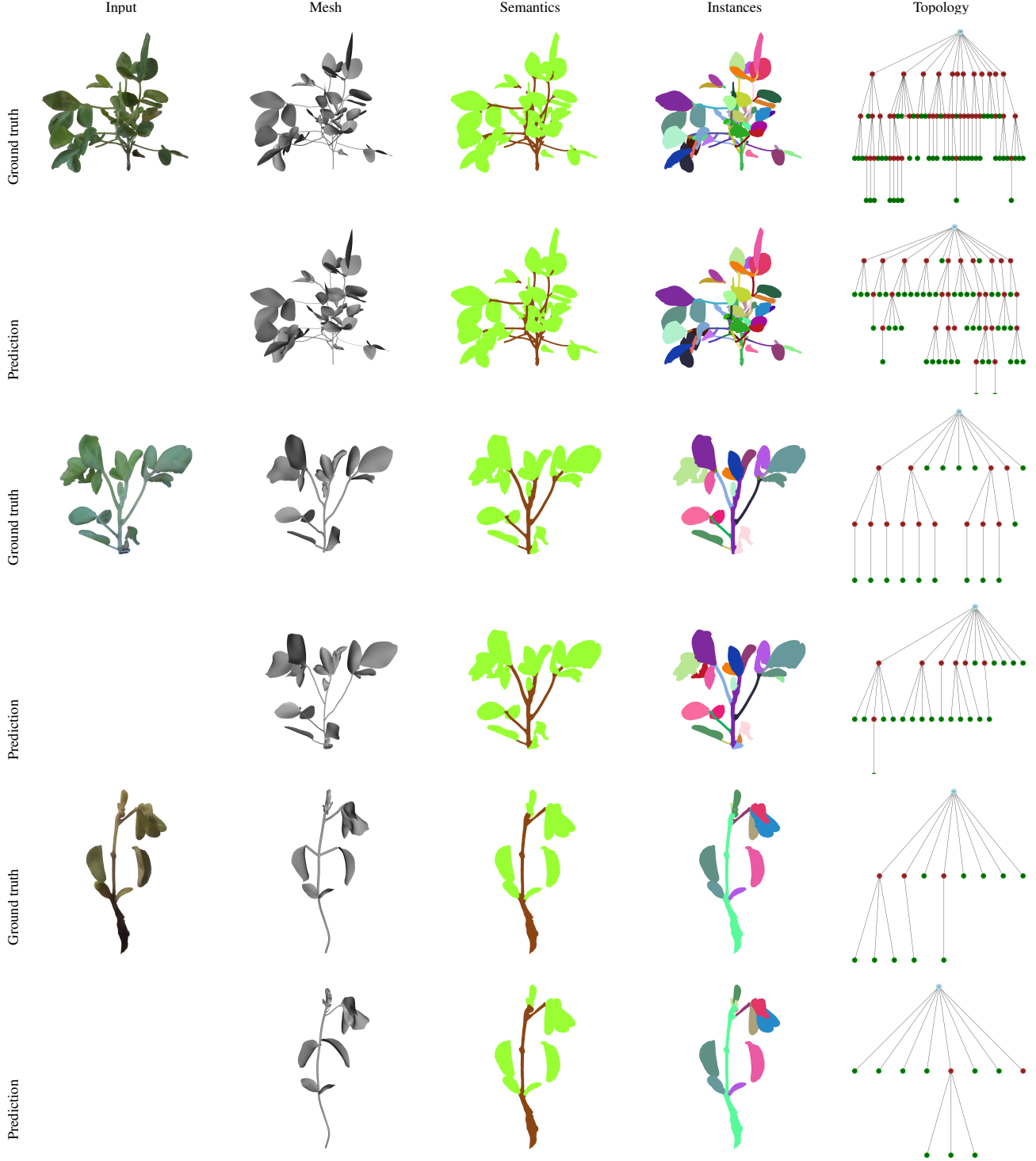
Figure 11. **Qualitative result of point-based reconstruction.** Given the unlabeled 3D point cloud as input, our model could faithfully recover the mesh, semantics, instances and topology.

parameters (Fig. 8), since there is no existing maize leaf scan dataset. Afterward, we apply the same pipeline as for soybeans to learn the 3D deformation basis and fit other parameters. The result (Fig. 6) shows that Demeter can faithfully capture maize plant shape.

F. Other Application

Agriculture We showcase this capability by generating small crop fields by placing fitted Demeter models in a grid, and passing them to Helios [1] to simulate the response of
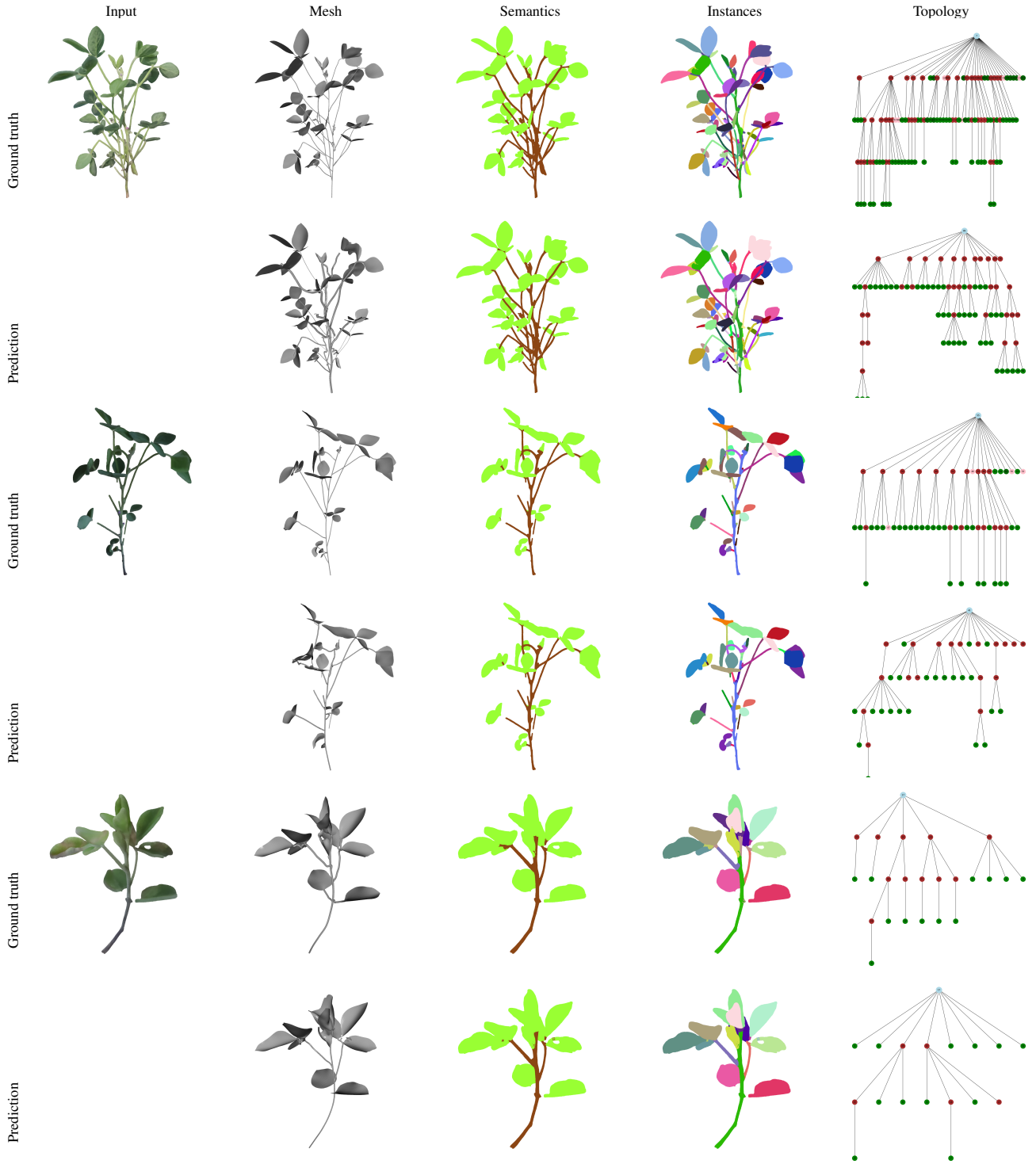
Figure 12. **Qualitative result of point-based reconstruction.** Given the unlabeled 3D point cloud as input, our model could faithfully recover the mesh, semantics, instances and topology.

the plant to weather variations over the course of a day. The weather variables were taken from data measured by a flux tower [2] and include temperature, humidity, radiation, and other environmental variables. In Fig. 14 and Fig. 15, we visualize two outputs of the simulation: photosynthesis rate

and stomatal conductance, which are both directly related to crop productivity.
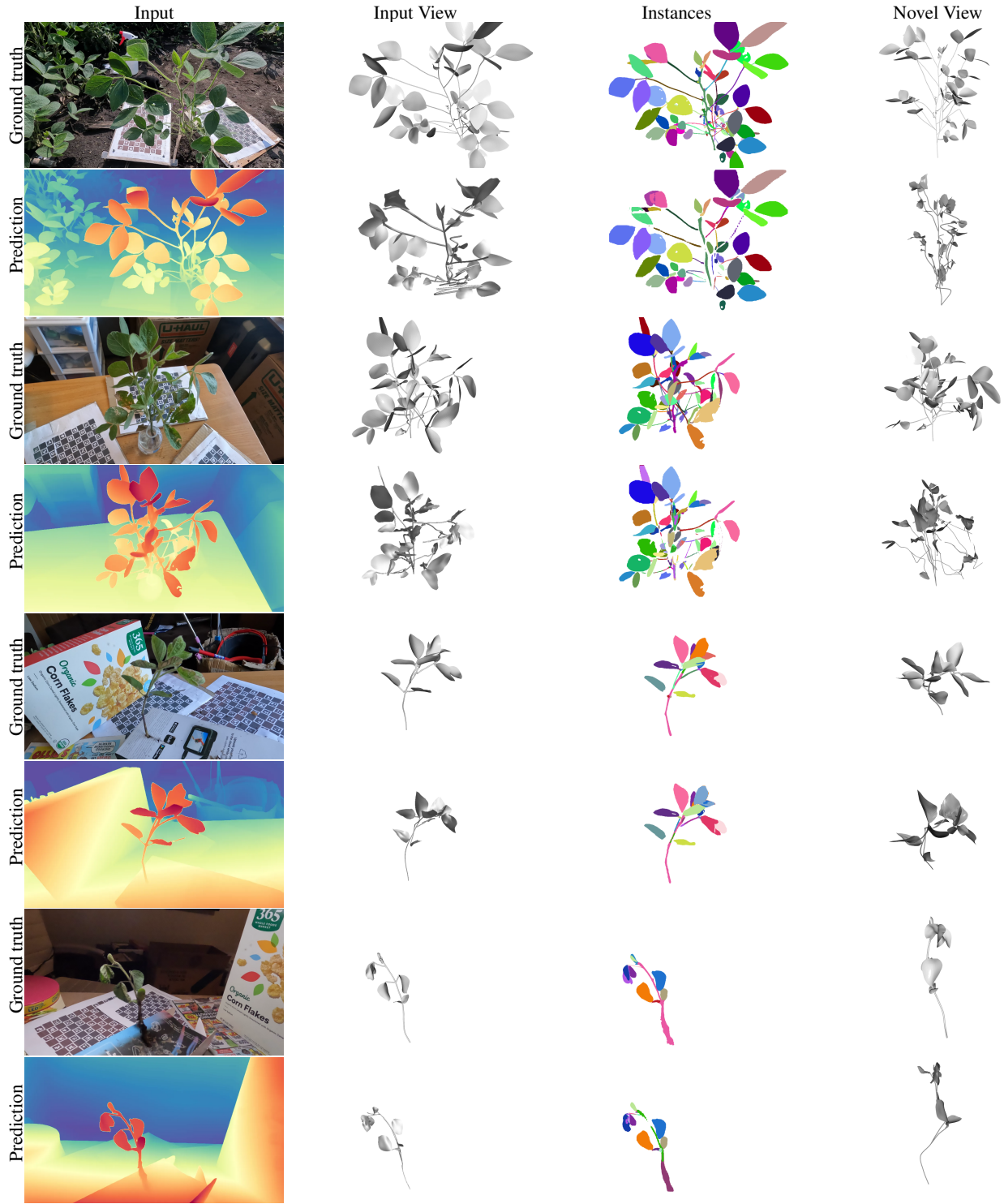
Figure 13. **Qualitative result of image-based reconstruction.** The depth prediction comes from off-the-shelf DepthAnything [5]

Rendering   By extracting texture from the bijective mapping mentioned in Eq. 6 in the main paper, we could add texture to the parametric model and using ray-tracing to obtain realistic rendering.

G. Discussion

The importance of disentanglement   In agriculture, disentangled and interpretable shape parameters are vital for phenotyping & genotyping, biophysical simulation, and
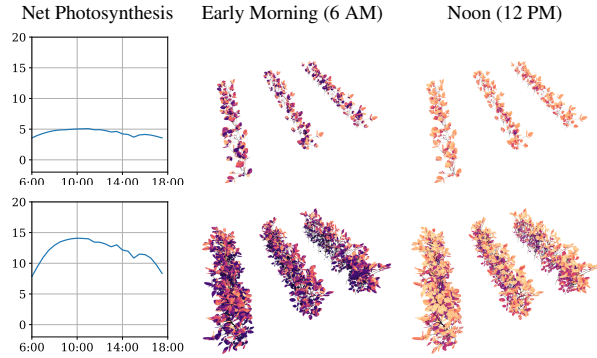
Figure 14. **Photosynthesis simulation results**. We perform simulations using Helios [1] on two soybean canopies generated by repeating Demeter-Soybean models. Left: timeseries of the net photosynthesis rate for the crop canopy over the course of a day, in units of μmolCO$_2$/m^2/s. Other columns: mesh visualization where each leaf face is colored according to the rate of photosynthesis over that face (brighter = higher rate).
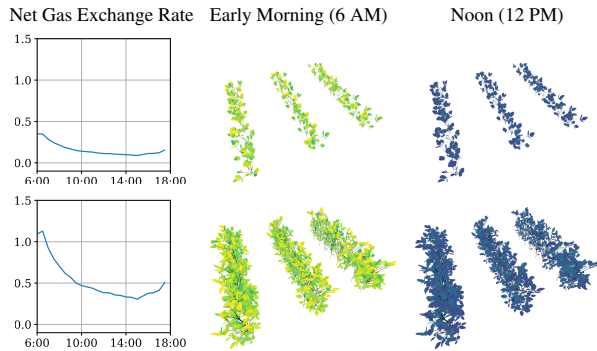

Figure 15. **Stomatal conductance simulation results**. Stomatal conductance is a measure of the degree of opening of a plant's stomata, which determines the rate of gas exchange (including carbon dioxide and water) between the plant and the air. Left: timeseries of the net gas exchange per unit ground area, in units of mol/m^2/s. Other columns: per-face rate-colored visualization.

process-models, as they serve as controllable, measurable variables in crop scientists' workflows. In graphics, disentangled shapes allows controllable procedural generation, texture mapping, and physically driven deformations.

Sensitivity to dataset size   The learnable parameters are PCA components for leaf and stem shape and deformation. Training these PCA requires only a handful of diverse plants per species—each plant contributes multiple leaves and stems—so long as the total # of leaves/stems exceeds the # of PCA control points per template. We can also adapt an initial template from related species (e.g., soybean to cowpea or tobacco) and finetune on even smaller datasets when annotated 3D scans are scarce. For example, our soybean model uses custom scans and FGLIR leaf data; Papaya is trained on PLANesT3D; tobacco adapts the soybean template and retrains on Plant3D, producing a viable

model from just 3–5 full 3D samples (with small fidelity trade-offs).

And training an instance-segmentation network from scratch typically requires more data (e.g., 50 plants), but fine-tuning a pretrained model on just a few examples generalizes well to new species.

Limitation   Although our model achieves realistic modeling, there are still some limitations. For example, the Demeter does not model skinning, so each part has uniform thickness and the connections between parts are unnatural.

Additionally, we have only demonstrated very basic capabilities for sample generation, such as copying the subtree from existing soybeans and pasting to random position and changing its topology. We believe that learning the distribution of the plant graph in latent space can better handle this task in the future.

References

[1] Brian N Bailey. Helios: A scalable 3d plant and environmental biophysical modeling framework. *Frontiers in Plant Science*, 10:1185, 2019. 6, 9

[2] Gilberto Pastorello, Carlo Trotta, Eleonora Canfora, Housen Chu, Danielle Christianson, You-Wei Cheah, Cristina Poindexter, Jiquan Chen, Abdelrahman Elbashandy, Marty Humphrey, et al. The fluxnet2015 dataset and the oneflux processing pipeline for eddy covariance data. *Scientific data*, 7(1):225, 2020. 7

[3] David Schunck, Federico Magistri, Radu Alexandru Rosu, André Cornelißen, Nived Chebrolu, Stefan Paulus, Jens Léon, Sven Behnke, Cyrill Stachniss, Heiner Kuhlmann, et al. Pheno4d: A spatio-temporal dataset of maize and tomato plant point clouds for phenotyping and advanced plant analysis. *Plos one*, 16(8):e0256340, 2021. 5

[4] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 5

[5] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv preprint arXiv:2406.09414*, 2024. 5, 8

[6] Albert J. Zhai, Xinlei Wang, Kaiyuan Li, Zhao Jiang, Junxiong Zhou, Sheng Wang, Zhenong Jin, Kaiyu Guan, and Shenlong Wang. Cropcraft: Inverse procedural modeling for 3d reconstruction of agricultural crops. *arXiv preprint arXiv:2411.09693*, 2024. 3

[7] Ying Zhang and Wei Su. Corn 3D point clouds. 2022. 4, 5