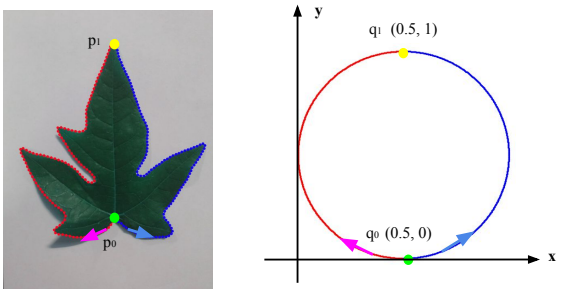
Demeter: A Parametric Model of Plant Morphology from the Real World

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 $\{\boldsymbol{p}\}_{i=0}^{K+2}$ 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 PerspectiveField 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
$\text{pa}(i)$	the parent of the node i
$\text{ans}(j)$	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
$\text{tp}(i)$	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.

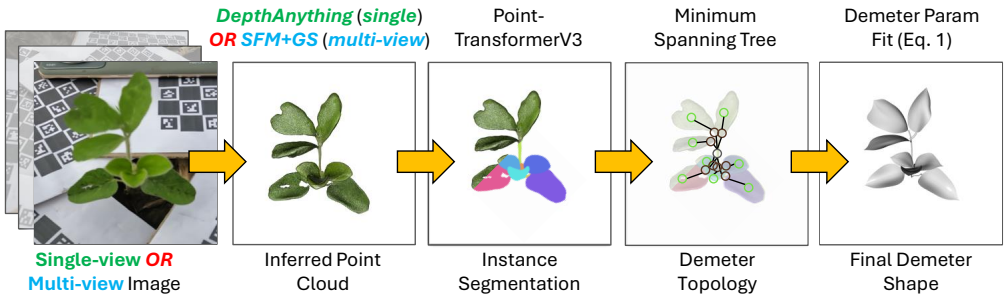


Figure 1. **Test-time Reconstruction Pipeline.** The pipeline transforms the input images to a segmented point cloud, extracts the topology and fits part templates, and finally outputs the parametric mesh.

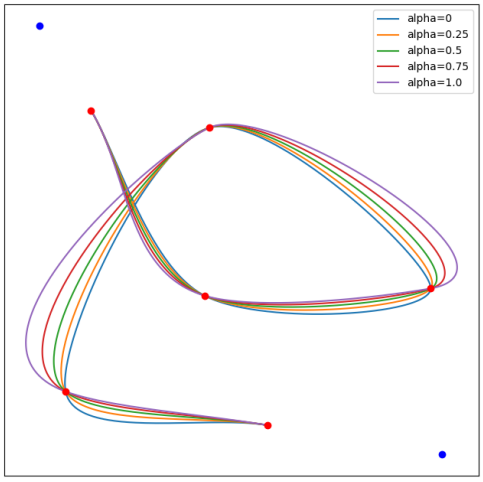


Figure 2. **Visualization of Catmull-Rom Curve.** The alpha parameter controls the curvature of the curve. We choose $\alpha = 0.5$ in this paper. The red dots represent the control points and the blue ones represent the virtual control points, which is necessary to draw the first and last segment.

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.

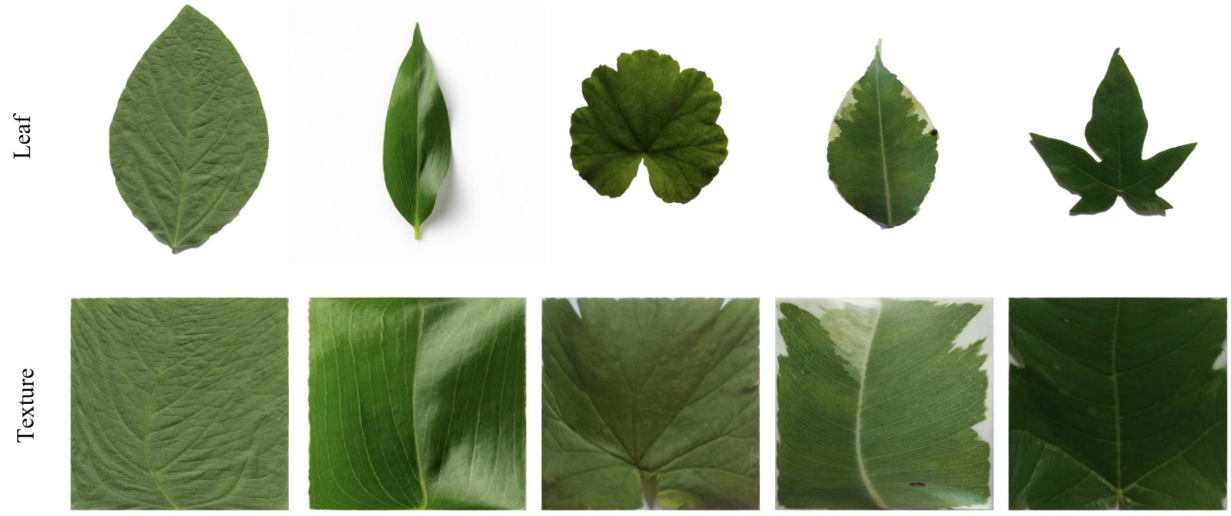


Figure 5. **The texture extracted from leaf images using bijective mapping.** Similarly, we could map the leaf to a square UV space and get the texture, which can be easily applied to the grid points of our parametric leaf.

	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-

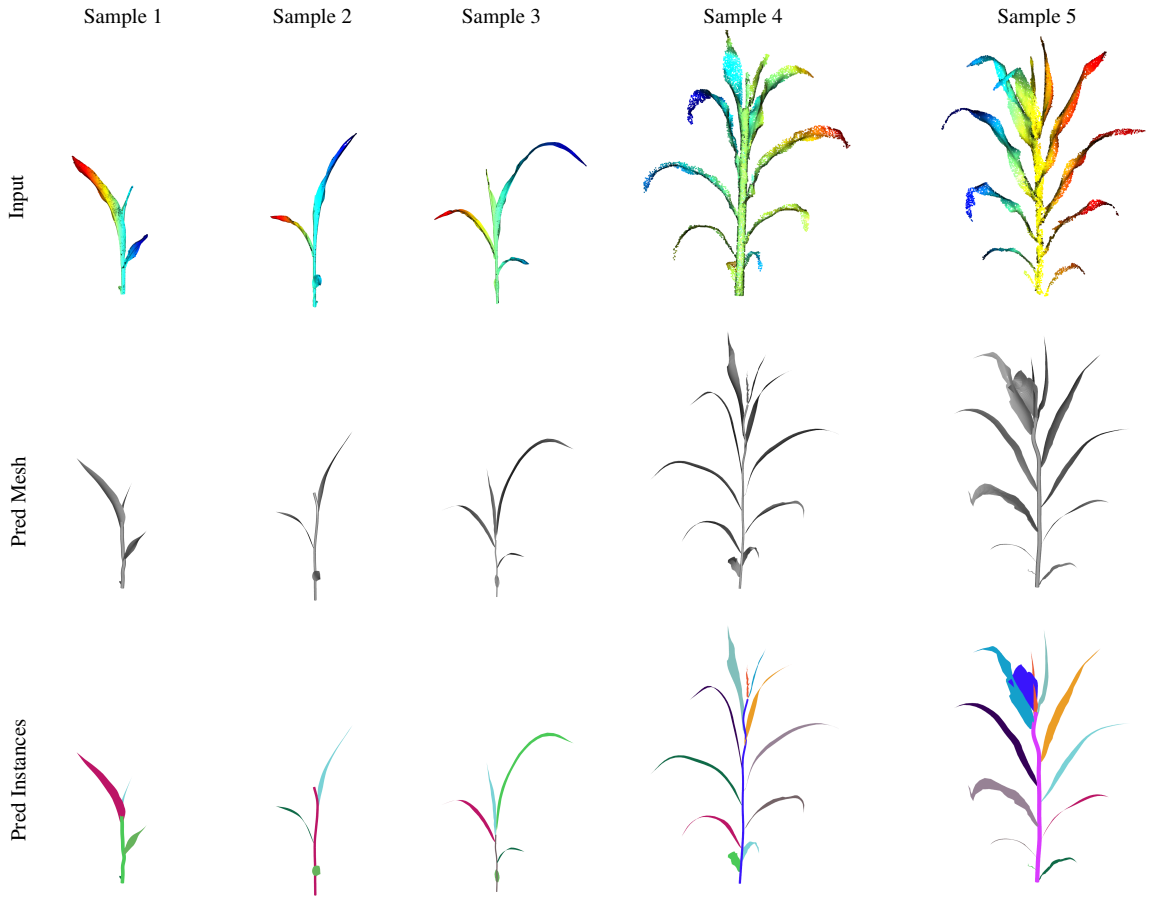


Figure 6. **Qualitative results of reconstructing maize.** We show the results of fitting Demeter-Maize models to point clouds from the Pheno4D dataset (sample 1,2 and 3) and web dataset [7] (sample 4 and sample 5). Different colors represent different instances. The results show that our model can generalize to species beyond soybeans and capture shape details.

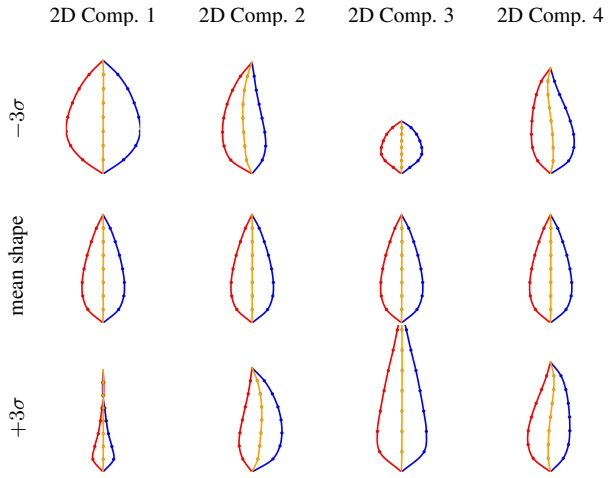


Figure 7. **PCA coefficient of soybean leaf.** We visualize the first several principle component of leaf for soybean in $(-3\sigma, 3\sigma)$.

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

3D Reconstruction Given the ground-truth instance segmentation, 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.

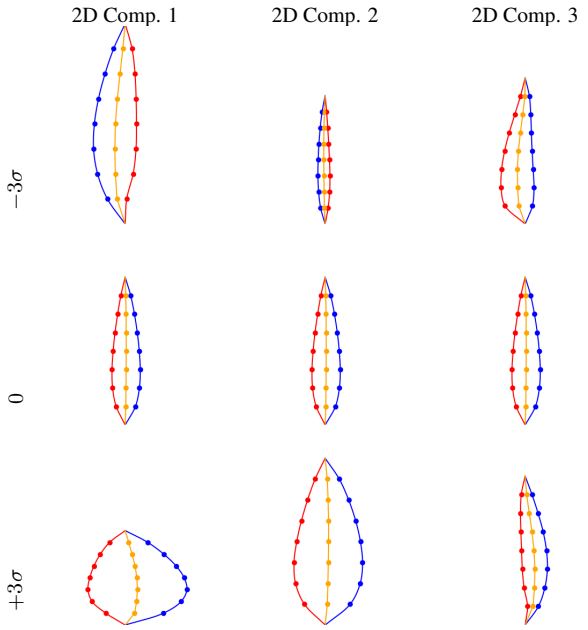


Figure 8. **PCA coefficients of Demeter-Maize leaf shape.** We vary the PCA coefficients for maize leaf shape across $(-3\sigma, 3\sigma)$ with respect to the mean shape for all 3 components. The red and blue points represent the control points for the 2D contour and the orange curve represents the initial skeleton for 3D deformation.

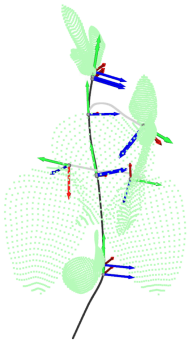


Figure 9. **Local coordinate system on the stem.** Each point on the curve is defined in a local coordinate system determined by the curve's intrinsic geometry at that specific point.

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 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.

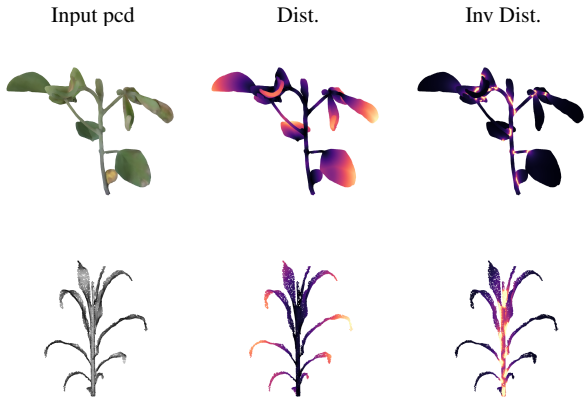


Figure 10. **Truncated Inverse Distance.** Given a input point cloud with instance segmentation, we select the point in every instance and calculate the distance to all other instances for that point (Middle). Afterwards, we select a threshold and calculate the inverse distance and train PointTransformer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

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

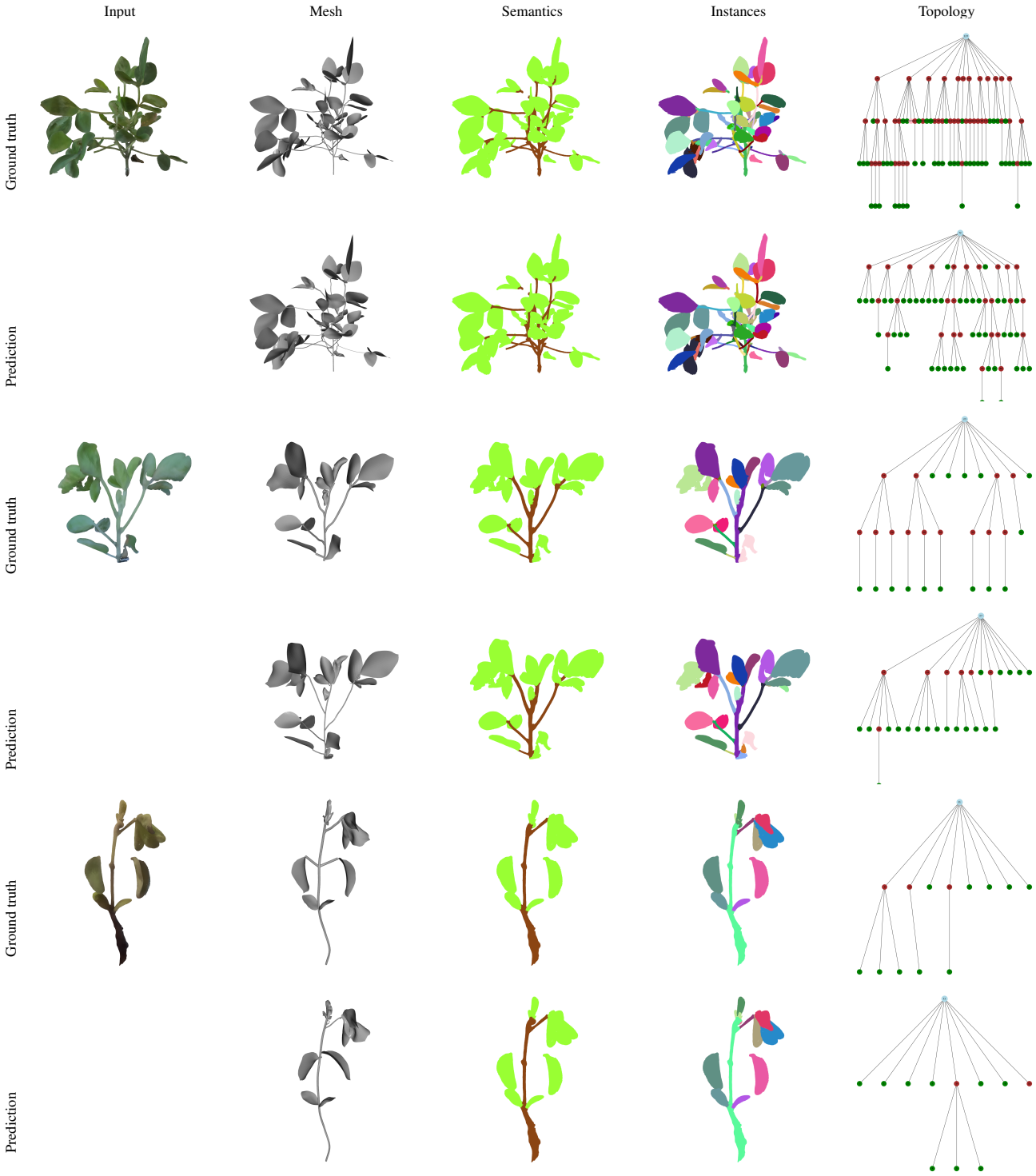


Figure 11. **Qualitative result of point-based reconstruction.** Given the unlabeled 3D point cloud as input, our model could faithfully recover the mesh, semantics, instances and topology.

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

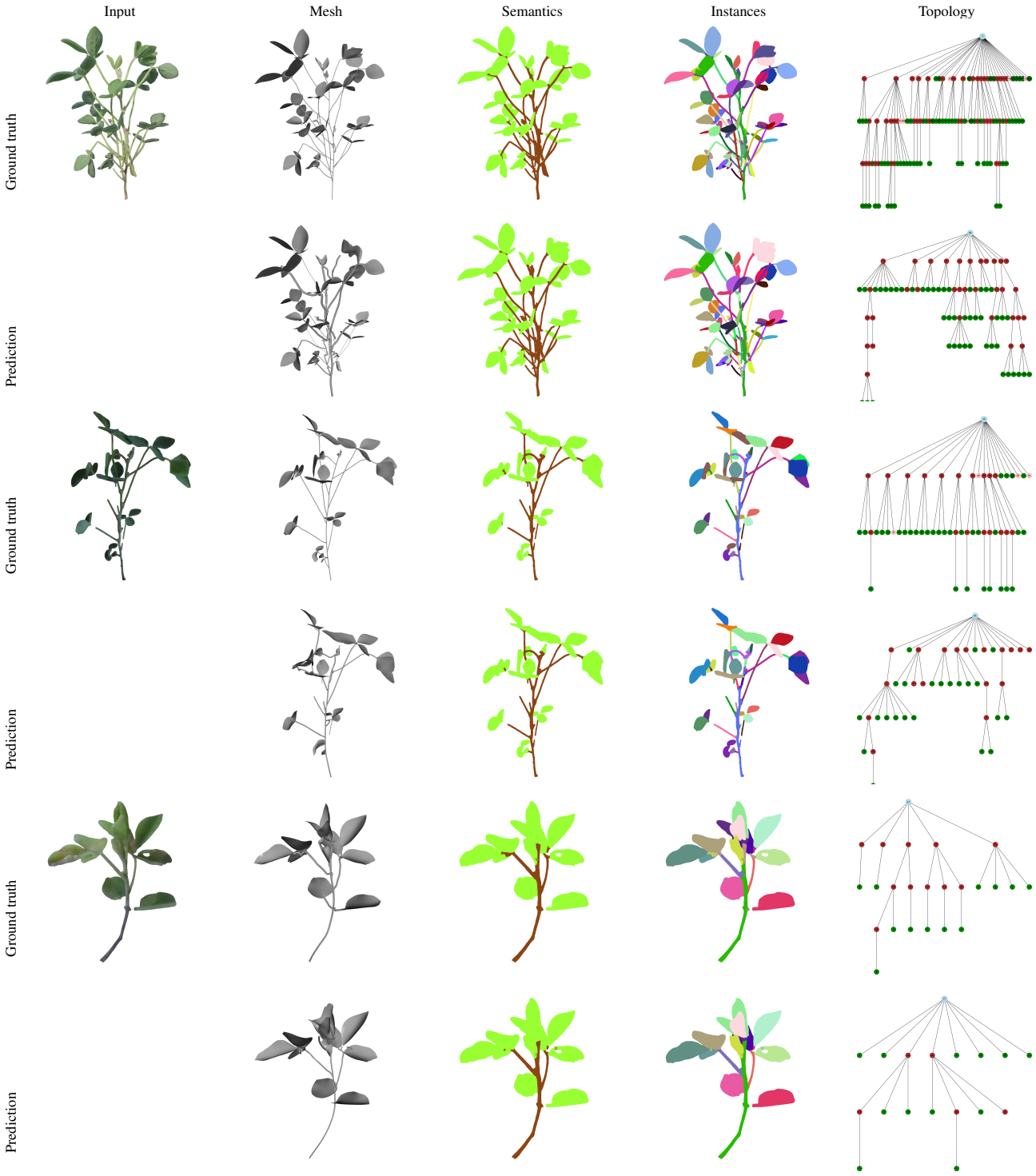


Figure 12. **Qualitative result of point-based reconstruction.** Given the unlabeled 3D point cloud as input, our model could faithfully recover the mesh, semantics, instances and topology.

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.

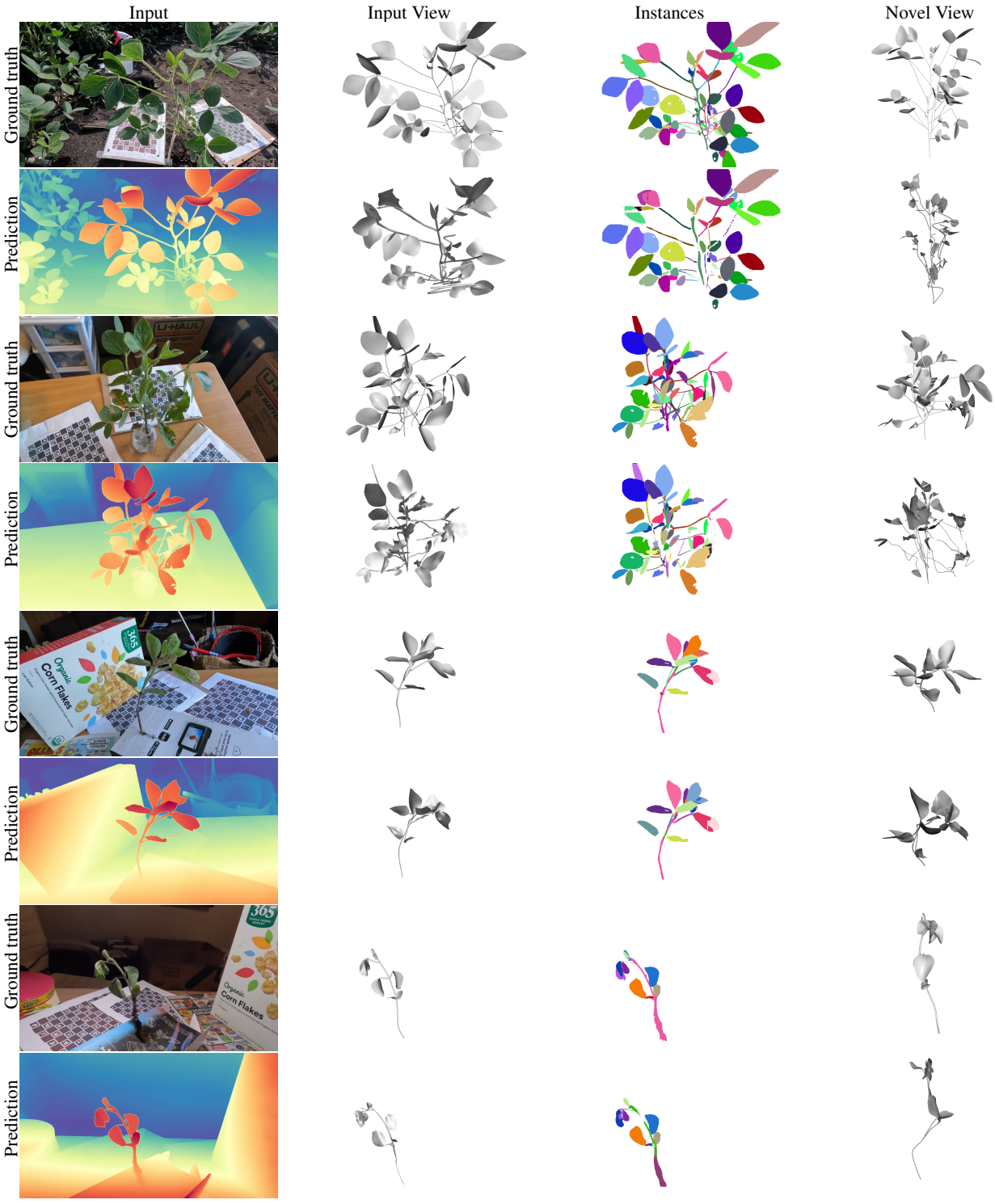


Figure 13. **Qualitative result of image-based reconstruction.** The depth prediction comes from off-the-shelf DepthAnything [5]

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

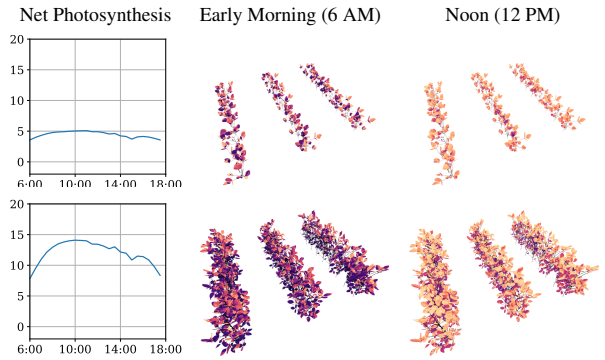


Figure 14. **Photosynthesis simulation results**. We perform simulations using Helios [1] on two soybean canopies generated by repeating Demeter-Soybean models. Left: timeseries of the net photosynthesis rate for the crop canopy over the course of a day, in units of μmolCO$_2$/m^2/s. Other columns: mesh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

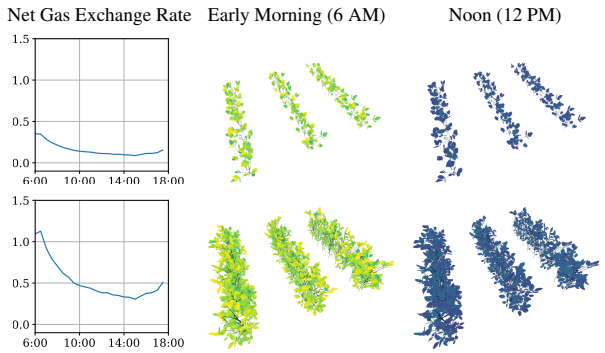


Figure 15. **Stomatal conductance simulation results**. Stomatal conductance is a measure of the degree of opening of a plant's stomata, which determines the rate of gas exchange (including carbon dioxide and water) between the plant and the air. Left: timeseries of the net gas exchange per unit ground 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