

Data-driven Flower Petal Modeling with Botany Priors

Chenxi Zhang

Mao Ye

Bo Fu

Ruigang Yang

Center for Visualization & Virtual Environments
University of Kentucky

Abstract

In this paper we focus on the 3D modeling of flower, in particular the petals. The complex structure, severe occlusions, and wide variations make the reconstruction of their 3D models a challenging task. Therefore, even though the flower is the most distinctive part of a plant, there has been little modeling study devoted to it. We overcome these challenges by combining data driven modeling techniques with domain knowledge from botany. Taking a 3D point cloud of an input flower scanned from a single view, our method starts with a level-set based segmentation of each individual petal, using both appearance and 3D information. Each segmented petal is then fitted with a scale-invariant morphable petal shape model, which is constructed from individually scanned exemplar petals. Novel constraints based on botany studies, such as the number and spatial layout of petals, are incorporated into the fitting process for realistically reconstructing occluded regions and maintaining correct 3D spatial relations. Finally, the reconstructed petal shape is texture mapped using the registered color images, with occluded regions filled in by content from visible ones. Experiments show that our approach can obtain realistic modeling of flowers even with severe occlusions and large shape/size variations.

1. Introduction

Plants modeling is one of the most difficult tasks in computer vision and graphics community because of their complex geometry and appearance. Flower, as the most distinctive part of a plant, has fine structures and wide variations, which makes reconstructing their 3D models a challenging task. Existing 3D modeling techniques for plants and vegetation are usually designed for large scale structures, such as trees, foliage, or based on pure synthesis given some pre-defined rules and templates.

The biggest challenge for flower modeling is occlusion. The tight formation of flower petals make segmentation and 3D reconstruction a very challenging task. In order to

make this modeling problem tractable, we develop a unique pipeline that incorporate domain-specific knowledge. More specifically, the shape space of petals (the most dominant components of a flower) can be learned from individually scanned petals and their relative spatial layout can be known a priori from botany study.

Our approach focuses on the parametric modeling of flower petals. It starts with the data capture process of a single flower, for which we use a structured light scanning system consisting of one camera and one projector to capture its shape in 3D. Once we capture the geometric details of a flower as a point cloud, our proposed method segments it into different components (petals) based on both 2D appearance and 3D depth information. Each segmented petal is then fitted using a scale-invariant morphable petal shape model built from individually scanned single petal samples. In our setup, flowers are captured from a single view (top view). The tightly overlaying flower petals make multi-view capture less effective. To handle the occlusions as well as to maintain the semantic layout of flowers, a set of novel constraints derived from botany studies and segmentation information are incorporated into the petal fitting process. Finally, the reconstructed flower model is texture mapped using the captured color images, and occluded regions are filled in with texture from other complete components. An overview of our approach is shown in Figure 1.

To our knowledge, our system is the first to focus on flower modeling, petals in particular, from 3D point cloud. The key contributions of our work can be summarized as: 1) a novel petal fitting algorithm that is robust to significant occlusions; 2) a robust scheme for flower petal segmentation, by extending a two-region level-set formulation to multiple regions; 3) a scale-invariant morphable petal shape model which can handle wider shape variations within a species, or even across species.

It should be emphasized that our reconstruction pipeline generates a *parametric model*, which is particularly suited for measurement, editing, and animation. For example, one could easily apply a geometric morphing between two models, or make global changes to the shapes by varying shape

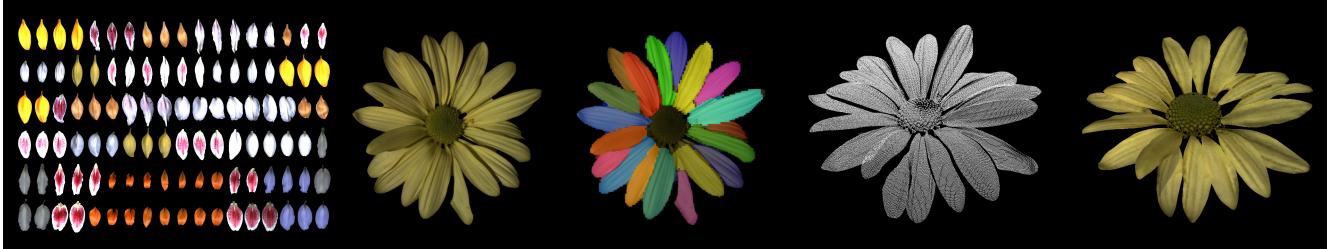


Figure 1. From left to right: 1) Petal database for Lily species; (2) Input image; (3) Petal segmentation; (4) Scanned 3D data; (5) Reconstructed 3D model.

parameters. While these are not explored in the context of this paper, we believe our approach will enable more research in the modeling and animation of an intrinsic class of objects, flowers, with applications in botany, entertainment, and visual simulations.

2. Related Work

Due to their importance in the real world, there are many approaches for modeling plants. They can be roughly divided into two categories: rule-based modeling and data-driven modeling. Rule-based methods use compact rules and grammars for building models of plants. As a prime work, a series of approaches based on the idea of L-system were developed [9, 16, 17, 18, 19]. The modeling of plant organs, such as leaves and petals, is a much less studied problem. Fowler and colleagues [4] developed a collision-based model for the spiral phyllotaxis effect, where plant organs are arranged in spiral patterns. Mundermann and collaborators [12] used leaf silhouettes to estimate leaf skeleton and further build leaf shape models. Fuhrer and colleagues [5] studied how to model and render small hairs on plants. Reunions and collaborators [21] developed procedural algorithms to model a number of leaf venation patterns. A related work in flower modeling is an interactive system by Ijiri and collaborators [6]. It has a graphical user interface for users to sketch flower models based on botanical constraints. However those work summarized above are purely rule-based, for which the realism and accuracy depend on the understanding of flower development and the effort of the modelers.

Recently with the proliferation of digital cameras and 3D scanning devices, there have been a number of data-driven approaches developed specific for tree, small plants, or foliage [20, 22, 14, 10, 11, 2]. Typically major tree branches are detected or interactively traced from 3D point clouds or images. Leaves are synthesized based on separate scanning, some heuristics, or mesh-fitting, so that the final model is visually similar to the input data. Approaches in this category aim to faithfully reconstruct the 3D model of plants based on the input. They usually focus on plants with a large number of leaves and the general structure of the whole plant. From algorithm perspective, the most related works

are from Quan [20] and Bradley [2]. Quan et al. use similar modeling procedures to ours, composed of an interactive leaf segmentation and template based model fitting. Their approach requires multi-view data in which the entire plant is captured, while in our case we only use data from a single view because multi-view data do not provide significant more converges due to the tight formation of flower petals. Both [20] and [2] use an exemplar leaf mesh to fit to the dense point clouds non-rigidly. [2] further learns a statistical model for continuing fitting other leaves, as well as for leaf synthesis when occlusions are too big. We instead require a shape database for flower petals to handle the significant occlusions in flowers.

After a survey of existing plant modeling techniques, we note that flowers, despite being the most significant focus of study for identification, are the least frequently studied, probably due to its complex structure and significant self-occlusions. Our method uses both a data-driven approach and knowledge in botany to handle these challenges.

3. Flower Petal Modeling

To capture the geometric details, we choose to use structured light scanner to acquire high quality 3D data of flowers. Our method starts from scanning individual flower petals with variations but from the same species, and building up a morphable model [1] for petal shape of a certain species. A level-set based active contour model is used for accurately segmenting the 2D image and 3D scanned points of a whole flower into different components(petals). Both 2D appearance and 3D depth information are used to guide the segmentation. Each segmented petal point clouds is subsequently fitted using the morphable model. We propose a joint multiple petal fitting algorithm using prior knowledge from Botany about flower spatial layout. Finally, the registered color image is used to generate texture maps for the 3D model. We will illustrate each part in details in next several sections.

3.1. Scale-invariant Morphable Model

We choose to use a learned morphable shape model to reconstruct flower petals because of the parametric nature of the model representation. The benefit of using morphable

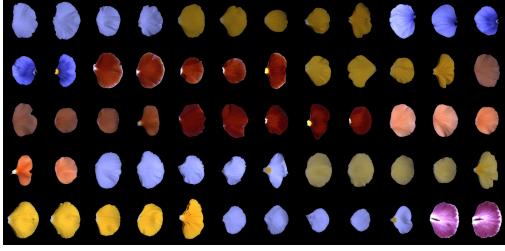


Figure 2. Petal database for Pansies(60 exemplars).

model is that the optimization affects the entire petal, as opposed to per-vertex based deformation method, therefore can robustly handle occlusions. The uniform parametric form of reconstructed shapes will benefit many future work, such as modeling the development of flowers.

To build the morphable model, we scan a collection of exemplar petals from the same species, but with noticeable variations. Figure 1 left shows one database of Lily with 108 exemplars while Figure 2 shows a Pansie database with 60 exemplars. The shape and size variations are not negligible even within the same species. To build the morphable model, we firstly align each exemplar shape to a reference shape via two principal axes. Then a non-rigid alignment is performed using CPD [13] to deform each exemplar shape to best fit the reference shape, in which way we obtain the correspondences. The correspondences are used to estimate a similarity transformation to transform each shape to reference shape coordinate system. Then we sample approximately 3500 vertices on the reference shape, and represent the shape of a petal by a shape-vector $S = (\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n) \in \mathcal{R}^{3n}$, where each \mathbf{v}_k is a three-dimensional vector representing 3D coordinate. The correspondences in transformed exemplar shapes are used to build a morphable shape model using Principle Components Analysis (PCA), defined as follows:

$$S_{model} = \bar{S} + \sum_{j=1}^{m-1} \alpha_j s_j = \bar{S} + \mathbf{B}\boldsymbol{\alpha} \quad (1)$$

where \bar{S} is the average of m exemplar shapes and $\mathbf{B} = (s_1, \dots, s_{m-1})$ are the eigenvectors of the covariance matrices defining petal shape space. $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_{m-1})$ are the coefficients of basis shapes.

Different from traditional method for building the morphable model, we compute a scale factor when transforming exemplar shapes to reference shape. This scheme is designed to eliminate the size variation among exemplar shapes, but focus more on the statistics of shape variations for reconstructing the details of a petal. The mean shape in morphable model is always firstly scaled to match the size of the source shape in our petal fitting process(Sec. 3.3). In this way, we build up a *scale-invariant* morphable model that can be applied to reconstruct petals of tremendously different sizes.

3.2. Flower Petal Segmentation

There has been some work on segmenting whole flowers from a scene [15], but few has been done on segmenting each individual component (petal) of a flower. The main challenges are from the high appearance similarity and noticeable self-occlusions, which makes the segmentation very challenging. Therefore, we manually specify a central position on each petal as an initialization to guide the segmentation.

We apply distance regularized level set evolution [8] formulation to an active contour model [7] to solve the petal segmentation problem. Both 2D and 3D gradient information are embedded in the active contour model as guidance for segmentation boundary evolutions. We extend the two-region level set method to multiple regions by defining p level set functions(LSF) $\phi_i, i \in (1, \dots, p)$, where p is the number of petals in a flower. Each LSF ϕ_i represents a region Ω_i , by setting $\Omega_i(x) < 0$ when $x \in \Omega_i$; $\Omega_i(x) > 0$ when $x \notin \Omega_i$; $\Omega_i(x) = 0$ when x is on the contour of Ω_i .

Let I be the color image and D be the depth map projected from 3D scanned data. We define a 2D gradient indicator function g_c and a 3D gradient indicator function g_d as

$$g_c = \frac{1}{1 + |\nabla G_\sigma * I|^2}; \quad g_d = \frac{1}{1 + |\nabla G_\sigma * D|^2} \quad (2)$$

Our final gradient indicator g is computed as a linear combination of g_c and g_d

$$g = \beta g_c + (1 - \beta) g_d \quad (3)$$

where G_σ is a Gaussian kernal for smoothing the color and depth image to reduce noise. For each LSF ϕ_i , we define an energy function $\mathcal{E}(\phi_i)$ by

$$\mathcal{E}(\phi_i) = \lambda \mathcal{L}_g(\phi_i) + \alpha \mathcal{A}_g(\phi_i) + \mu \mathcal{R}_{\hat{p}}(\phi_i) \quad (4)$$

By finding the minimum of $\mathcal{E}(\phi_i)$, we can obtain the segmentation as the region Ω_i that $\phi_i < 0$ represents. λ, α are the coefficients of the energy functions $\mathcal{L}_g(\phi_i)$ and $\mathcal{A}_g(\phi_i)$. μ is the coefficient of distance regularization term $\mathcal{R}_{\hat{p}}(\phi_i)$. They are defined as:

$$\mathcal{L}_g(\phi_i) = \int_{\Omega_i} g \delta(\phi_i) |\nabla \phi_i| dx \quad (5)$$

$$\mathcal{A}_g(\phi_i) = \int_{\Omega_i} g H(-\phi_i) dx \quad (6)$$

$$\mathcal{R}_{\hat{p}}(\phi_i) = \int_{\Omega_i} \hat{p}(|\nabla \phi_i|) dx \quad (7)$$

where δ and H are Dirac delta function and Heaviside function, \hat{p} is a potential function for distance regularization.

The energy in $\mathcal{L}_g(\phi_i)$ computes the line integral of the function g along the zero level contour of ϕ_i , which is minimized when the zero level contour of ϕ_i is located at the petal boundary indicated by g . The energy $\mathcal{A}_g(\phi_i)$ computes the weighted area of region where $\phi_i(x) < 0$. It is

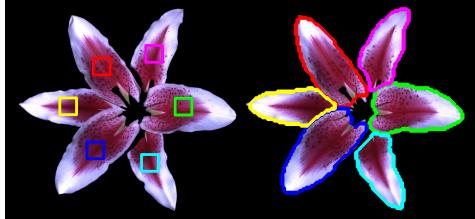


Figure 3. Left: Initial segmentation. Right: Final segmentation.

used to accelerate the movement of zero level contour in the level set evolution process, while slowing down when it arrives at petal boundaries where g takes smaller values. The distance regularization term $\mathcal{R}_p(\phi_i)$ is defined for maintaining the signed distance property of LSF.

The energy function in Eq. 4 can be minimized by solving the following gradient flow:

$$\frac{\partial \phi_i}{\partial t} = \delta(\phi_i) \left(\lambda \operatorname{div} \left(g \frac{\nabla \phi_i}{|\nabla \phi_i|} \right) + \alpha g \right) + \mu \operatorname{div}(d_{\hat{p}}(\nabla \phi_i) \nabla \phi_i) \quad (8)$$

Minimization of Eq. 4 is under the constraint $\bigcup_i \Omega_i = \Omega$ and $\bigcap_i \Omega_i = \Phi$, namely, we want to prevent overlapped and vacuum regions. Therefore we employ the idea from [3] to enhance the evolution process of ϕ_i as:

$$e_k := \lambda \operatorname{div} \left(g \frac{\nabla \phi_i}{|\nabla \phi_i|} \right) + \alpha g \quad (9)$$

$$\begin{aligned} \frac{\partial \phi_i}{\partial t} &= \delta(\phi_i) \left(e_i - \min_{\delta(\phi_j) > 0; j \neq i} (e_j, e_i - 1) \right) \\ &\quad + \mu \operatorname{div}(d_{\hat{p}}(\nabla \phi_i) \nabla \phi_i) \end{aligned} \quad (10)$$

We initialize each LSF with a binary step function ϕ_i^0 defined by

$$\phi_i^0(x) = \begin{cases} -c & \text{if } x \in R \\ c & \text{otherwise} \end{cases} \quad (11)$$

where $c > 0$ is a constant, and R is a square region centered at the initial position manually clicked on each petal. As shown in Figure 3, the initialized regions finally evolve to accurately match the boundary of each petal, with no overlap or vacuum.

3.3. Flower Petal Fitting

To handle the occlusions and maintain correct 3D spatial relations of multiple flower petals, we propose a joint petal fitting scheme, incorporating prior constraints from spatial layout information and segmentation results. It is worth noting that the input petal shape and the morphable shape model are in different coordinate systems. Therefore, we estimate a similarity transformation (s, R, t) between the two coordinate systems that transforms reconstructed shape from morphable model space to input space for fitting. Specifically, suppose a flower has p petals and L different layers. Let $L(i)$ be the layer where i^{th} petal

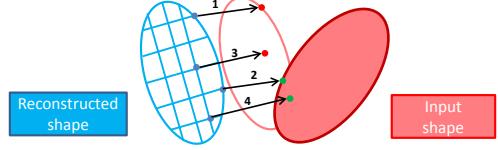


Figure 4. Four cases of finding correspondences on input shape. Red points stand for true correspondences, and green points are false correspondences.

lies in. We minimize the following energy over the set of model parameters $\vec{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_i, \dots, \alpha_p)$, where $\alpha_i = (\alpha_i^1, \alpha_i^2, \dots, \alpha_i^{m-1})$ is the shape coefficients of i^{th} petal. Suppose (s_i, R_i, t_i) are the similarity transformation from morphable model space to the i^{th} petal.

$$E(\vec{\alpha}) = \lambda_p E_P(\vec{\alpha}) + \lambda_c E_C(\vec{\alpha}) + \lambda_s E_S(\vec{\alpha}) \quad (12)$$

There are three terms in this energy function, which are defined as:

$$E_P(\vec{\alpha}) = \sum_{i=1}^p \|\mathcal{W}_i(s_i R_i(\bar{S} + \mathbf{B}\alpha_i) + \mathbf{t}_i - C_i)\|^2 \quad (13)$$

$$\begin{aligned} E_C(\vec{\alpha}) &= \sum_{i=1}^{p-1} \|(s_i R_i(\bar{S} + \mathbf{B}\alpha_i) + \mathbf{t}_i)_{(k)} \\ &\quad - (s_{i+1} R_{i+1}(\bar{S} + \mathbf{B}\alpha_{i+1}) + \mathbf{t}_{i+1})_{(k)}\|^2 \end{aligned} \quad (14)$$

$$E_S(\vec{\alpha}) = \sum_{L(r)=L(s)} \|\alpha_r - \alpha_s\|^2 \quad (15)$$

The first term $E_P(\vec{\alpha})$ measures the distance between reconstructed model and the set of all correspondences C_i on i^{th} petal. $\mathcal{W}_i = \operatorname{diag}(w_i^1, \dots, w_i^n) \otimes I_3$ is the weight matrix for all vertices $v_i = (v_i^1, \dots, v_i^n)$ in the shape S_i . I_3 is 3×3 identity matrix. When finding correspondences, we enforce boundary-to-boundary, and inner-to-inner matching between reconstructed shape and input. We also compute an occlusion map (occluded region) and occluded boundary (false boundary) for each input petal based on petal segmentation and layer information from Botany. Therefore, there are four cases that each vertex v_i^k can be related to its correspondence $C_i(v_i^k)$ on input during reconstruction: 1) v_i^k is on boundary of shape model and $C_i(v_i^k)$ is also on real boundary in input; 2) v_i^k is on boundary of shape model but $C_i(v_i^k)$ is on false boundary in input; 3) v_i^k is inside the petal model and $C_i(v_i^k)$ is not occluded in input; 4) v_i^k is inside petal model but $C_i(v_i^k)$ is occluded in input. For case 1, we set $w_i^k = w_b$; for case 3, we set $w_i^k = w_{nb}$; for case 2 and 4, we set $w_i^k = 0$. For a vertex that $\|v_i^k - C_i(v_i^k)\| > \tau$, we set $w_i^k = 0$. Figure 4 shows the four cases when finding correspondences on input shape.

The second term $E_C(\vec{\alpha})$ enforces the root of each reconstructed petals to converge to the same point in 3D space. The subscript (k) stands for a pre-defined root vertex index in the morphable shape model. This is a reasonable semantic prior for many types of flowers in real world. Captured

from top view, there is always missing data around the root region of each petal, due to the occlusion caused by pistil. Therefore, adding this prior can effectively assist the convergence of petal roots in 3D space, which also contributes to more realistic reconstructed flowers as a whole.

The last term $E_S(\vec{\alpha})$ encodes the similarity of different petals on the same flower. It enforces petal r and s on the same layer having similar shapes, modeled by the Euclidean difference of coefficient vectors α_r and α_s . This term can effectively add strong shape priors to petals with severe occlusion, by assuming that it has similar shape with less-occluded petal in the same layer.

The optimization of our cost function $E(\vec{\alpha})$ is subject to two further constraints. In a reconstructed model, multiple petals in overlapped regions should maintain the same 3D spatial relations as in input scans. We therefore induce a constraint that the reconstructed depth of occluded regions should be larger than the depth of the part in another petal that occludes it. By projecting the reconstructed shape to image, we identify the vertices that lie in the occluded region of that petal. From the segmentation result, we also know which petals are occluding these vertices, as well as their corresponding reconstructed depth values.

The other constraint restricts the reconstructed shape to lie in our training sample spaces. These two constraints are reasonably defined as:

$$(s_i R_i (\bar{S} + \mathbf{B}\alpha_i) + \mathbf{t}_i)_z \geq d_o \quad \forall k \in O_i \quad (16)$$

$$\mu - b\sigma \leq \alpha_i \leq \mu + b\sigma \quad (17)$$

where O_i represents the set of vertex indices in occluded regions of the i^{th} petal. The superscript z stands for depth(z-coordinate) value of a vertex. d_o is the corresponding depth value in the occluding petal. μ and σ are the means and standard deviations of the coefficients of training samples, and $b > 0$ is a constant.

We optimize the cost function $E(\vec{\alpha})$ iteratively in a coarse-to-fine fashion. In the first stage, we set the weights of inner vertices in $E_P(\vec{\alpha})$ to 0, namely, $w_{nb} = 0$, only align the boundary vertices of input and reconstructed shape, and only subject to constraint in Eq. 17. After boundary points converges, w_{nb} is restored in the following optimization process for better fitting inner regions. We find correspondences C_i on input shape using the 2D projections, given the boundaries are well aligned. Finally we incorporate constraint in Eq. 16 to refine the relative depth relations among different petals. Each stage is repeated until convergence, or reaching a maximum number of iterations N . Intermediate results of each stage showing progressively improvement can be seen in Figure 5.

Simultaneous optimization of $E(\vec{\alpha})$ over $\vec{\alpha}$ and (s_i, R_i, \mathbf{t}_i) is non-linear. For simplicity, we linearly optimize over $\vec{\alpha}$ and (s_i, R_i, \mathbf{t}_i) separately in each iteration. We

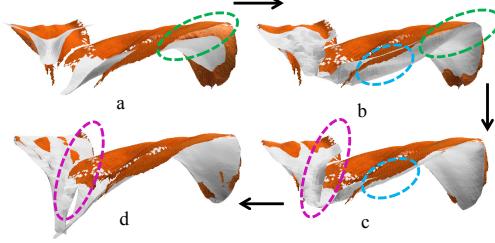


Figure 5. Intermediate results of coarse-to-fine petal fitting. a) After initial alignment; b) After boundary alignment; c) After fitting with inner points; 4) After adding relative depth constraints.

initialize $\vec{\alpha} = \mathbf{0}$, namely, using the mean shape \bar{S} as starting point for all petals. In order to estimate the initial similarity transformation $(s_i^0, R_i^0, \mathbf{t}_i^0)$, we use petals with sufficient visibility based on segmentation information to estimate an average target size. Besides, a rough root position is estimated via the optimal convergence point of their principal axes. For each pedal, the mean shape is then attached to the root, then aligned with the corresponding principal axes, and finally scaled to the target size. In each iteration, (s_i, R_i, \mathbf{t}_i) are firstly re-estimated before optimizing $E(\vec{\alpha})$ over $\vec{\alpha}$. Algorithm 1 shows the details of our fitting algorithm.

Figure 6 shows a challenging case with severe occlusion. Our joint multiple petal fitting algorithm can still successfully reconstruct the complete shape.

```

Initialization:  $k = 1; \vec{\alpha}^k = \mathbf{0};$ 
 $(s_i, R_i, \mathbf{t}_i) = (s_i^0, R_i^0, \mathbf{t}_i^0); \quad \forall i \in 1, 2, \dots, p$ 
for  $l = 1 : 3$  do
    while  $k \leq N$  do
         $S_i^k = \bar{S} + \mathbf{B}\alpha_i^k; \quad \forall i \in 1, 2, \dots, p$ 
        Compute  $(s_i, R_i, \mathbf{t}_i)$  from
         $S_i^k$  to  $i^{th}$  input petal;  $\forall i \in 1, 2, \dots, p$ 
        Find closet points  $C_i$ ;
        Compute a least-square solution of  $\vec{\alpha}^{k+1}$  for
        minimizing  $E(\vec{\alpha})$ ; s.t condition l
        if  $\|\vec{\alpha}^k - \vec{\alpha}^{k+1}\| \leq \epsilon$  then
            | break;
        else
            |  $\vec{\alpha}^k = \vec{\alpha}^{k+1}; k = k + 1;$ 
        end
    end
     $\vec{\alpha} = \vec{\alpha}^k;$ 
condition 1:  $w_{nb} = 0$  and Eq. 17;
condition 2: Eq. 17;
condition 3: Eq. 16 and Eq. 17;

```

Algorithm 1: Joint flower petal fitting algorithm

3.4. Texture Mapping

After reconstructing the shape of each petal, we use standard texture mapping to add texture to our parametric flower



Figure 6. Example of severely occluded petal reconstruction. From left to right: 1) Scanned petal data with occlusion; 2) Reconstructed petal without texture; 3)Reconstructed petal with texture.

model. Each petal is textured mapped individually, by projecting vertices in our parametric shape model on 2D images. For occluded regions, we fill in with content from non-occluded petals for synthesizing a complete texture for partially scanned input.

4. Experiments

We demonstrate our flower modeling algorithms on two different flower species, Lily and Pansie. The first species has large shape and size variations and the second has severe occlusions. We also demonstrate the scalability of our method for cross-species modeling, by reconstructing a third species, Dasiy, using the morphable model built from Lily. These two species share similar petal shapes but in remarkably different sizes. We use the following parameter settings to generate all results: $\{\beta, \lambda, \alpha, \mu, c, \lambda_p, \lambda_c, \lambda_s, w_b, w_{nb}, \tau, b, N, \epsilon\} = \{0.3, 5, -3, 0.2, 2, 1, 1000, 2, 80, 10, 15, 3, 20, 1e-3\}$.

The prior knowledge about Lily obtained from Botany is the two layer structure, each of which consists three petals. Each top layer petal occludes the bottom layer petals on its two sides. There are multiple variations in petal's shape, size and color across Lily species. Even petals on the same flower have very different shapes across layers. To test the effectiveness of our method, we use a single morphable shape model from Lily species to reconstruct samples of different variations.

Figure 7 shows the results of reconstructing four different variations of Lily. Despite noticeable occlusions and shape variations, our method still achieves high quality modeling of flower petals. Especially, we realistically recovers the shapes of occluded petals. It is worth mentioning that the orange Lily is about half the size of the other three variations. Our scale-invariant morphable model can robustly reconstruct the petal shapes by eliminating the size ambiguity and focus on shape variations.

The second species is Pansie which contains five petals on three layers, with top layer(one petal) occluding middle one(two petals), and middle layer occluding the bottom one(two petals). This type of flower is more challenging due to the severe occlusions in bottom layer. Only a tiny part of that layer is visible during scan. In the same way, we use one morphable shape model for this species to reconstruct samples of four different variations. As shown in

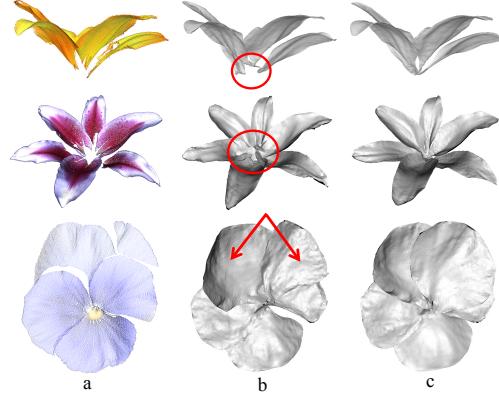


Figure 9. Constraint importance evaluation. From left to right: a) Scanned 3D data; b) Remove one constraint; c) With all constraints. From top to bottom: Comparisons with removing 1) root convergence constraint(E_C); 2) relative depth constraint(Eq. 16); 3) similarity constraint(E_S). Zoom-in for better visualization.

Figure 8, our algorithm reconstructs the flower models with high quality. For the two petals in bottom layer with severe occlusions, our method can fit the visible part very well, while maintaining reasonable predictions on invisible parts. In Pansie species, the depth differences between neighboring layers are very small. Such large occlusion regions allow a large degree of freedom when fitting invisible regions, which is prone to violate 3D relative geometries among different parts. Our relative depth constraints for fitting can efficiently avoid this and obtain more realistic reconstructions. The similarity constraints on petals in same layer constrain each other within a reasonable shape range during fitting and finally reach a common good solution.

Constraint evaluation To highlight the importance of our proposed constraints, we conduct comparisons with removing one constraint each. Figure 9 shows corresponding results. We can see our joint petal fitting scheme with these prior constraints are crucial in obtaining high quality modeling. As marked out in red, without the root convergence constraint E_C , roots of individually fitted petals usually cannot converge, making the reconstruction unnatural. This constraint also assists in handling occlusions by preventing petal from shrinking to only visible parts. The relative depth constraint(Eq. 16) maintains correct 3D geometry relations among different layers. The similarity constraint E_S can ensure a reasonable reconstructed shape when a petal is under severe occlusion.

We also make a comparison with a leaf fitting method in a state-of-the-art foliage reconstruction work [2], which use the same morphable shape model for leaf. They use similar framework but without any prior constraints. Figure 10 shows a comparison with their leaf fitting method. We can see that our method with the proposed constraints can better reconstruct occluded regions, recover correct 3D geometry

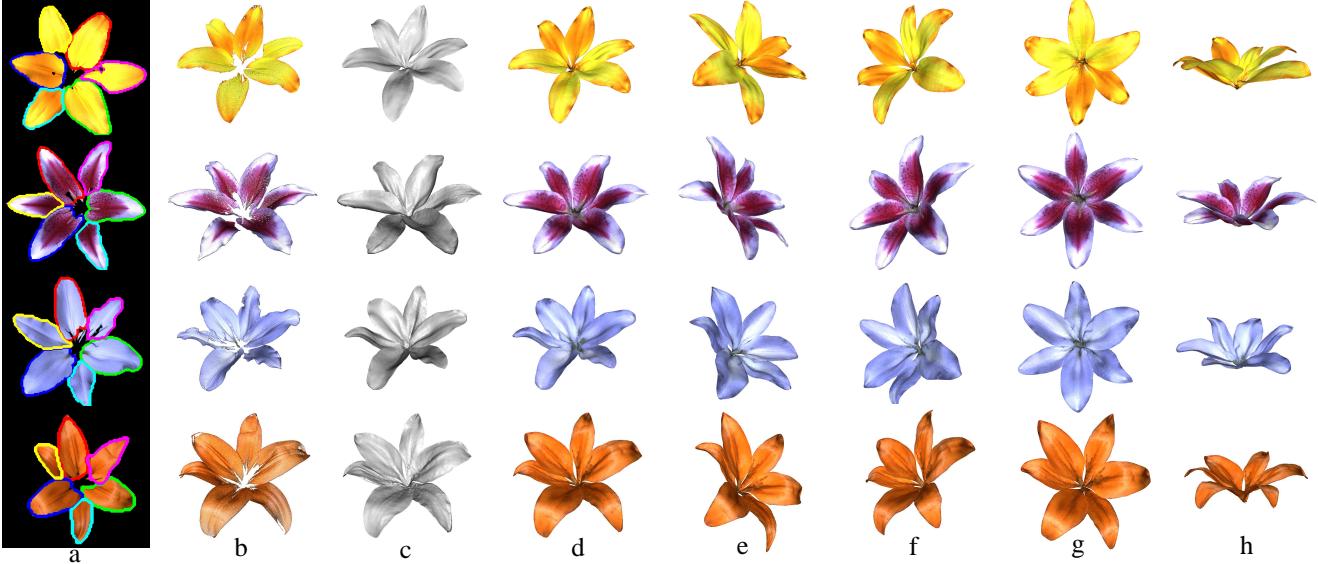


Figure 7. 3D modeling of Lily species. From left to right: (a) Petal segmentation; (b) Scanned 3D data; (c) Reconstructed model without texture; (d) Reconstructed model with texture; (e)-(f): Model from different views. The orange example(last row) is scaled up for visualization purpose. Zoom-in for better visualization.

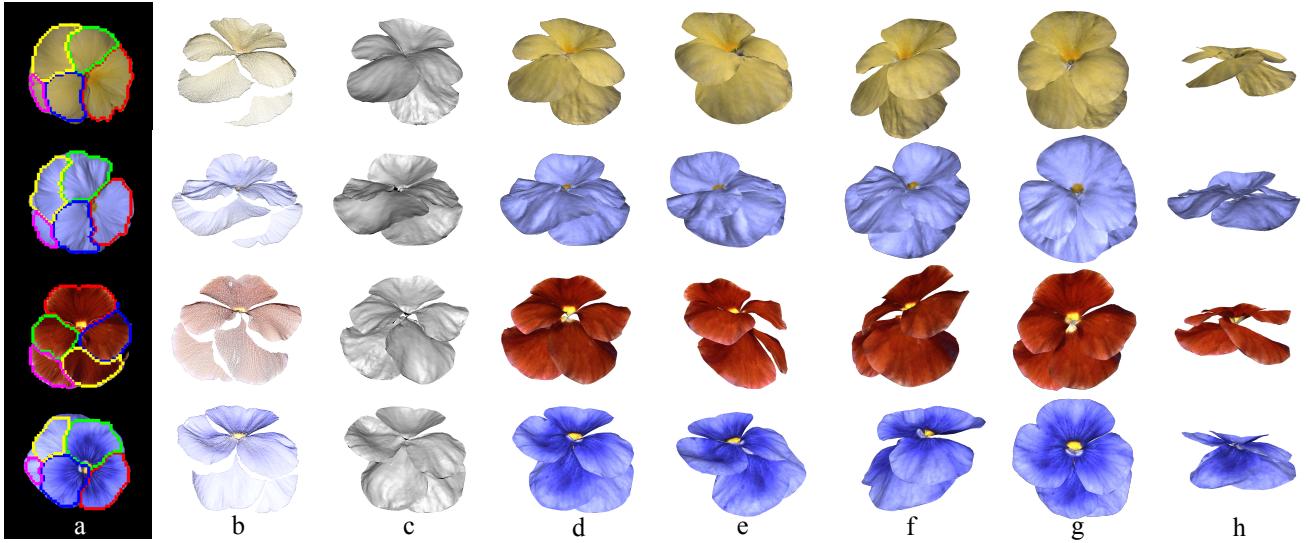


Figure 8. 3D modeling of Pansy species. From left to right: (a) Petal segmentation; (b) Scanned 3D data; (c) Reconstructed model without texture; (d) Reconstructed model with texture; (e)-(f): Model from different views. Zoom-in for better visualization.

relations of different components and more surface details. The reason that the leaf fitting method in [2] cannot be applied to our flower petal is that petals in our database have significantly larger variations in shape and size, compared to leaves they work on. Without additional constraints, the individually fitted shape has more freedom and is prone to grow or shrink to non-realistic shapes.

Cross species test Last but not the least, to demonstrate the scalability of our method, we conduct a cross species test, using morphable shape model of Lily species to reconstruct flowers from a different species(Daisy). The two

species share similar shape, but the size of Lily petal is approximately 20 times larger than Dasiy's. Reconstructing Dasiy is even more challenging since there are a large number of occlusion regions and relative 3D geometry relations to handle (approximate 20 petals). The cross species reconstruction result is shown in Figure 1. Despite substantial variations, our algorithm can still robustly obtain high quality reconstructions as long as the training and testing species share similar petal shapes.

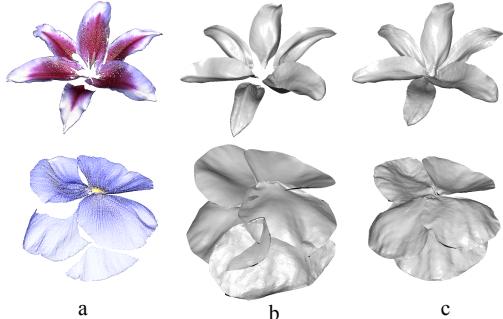


Figure 10. Comparison with leaf fitting method in [2]. From left to right: a) Scanned 3D data; b) Results from [2]; c) Our results. Zoom-in for better visualization.

5. Conclusions

In this paper we present a framework for 3D modeling of flower petals. Our approach builds a scale-invariant morphable model of flower petal shape from different variations within a species. In our data-driven modeling approach, the key idea is to use domain knowledge from botany study in petal fitting to handle occluded shape and maintain correct 3D spatial relations. We demonstrate our modeling algorithm with high quality reconstructions for various flower species.

A limitation of our current approach is that there are still inaccuracies in finding correspondence among variations of a flower species, which can downgrade the quality of morphable shape model. And the performance of our fitting algorithm is dependent on the segmented boundary of petals. More constraints might need to be incorporated for robust fitting under inaccurate petal boundaries. In addition, an alternative scanning method is to use volumetric imaging techniques (e.g. micro-CT scanners) which can lead to more complete data acquisition and 3D modeling.

A future extension of the work is to model the development process of flowers. By capturing real-world data for multiple flowers in different life stages, a reconstruction framework taking the temporal domain into consideration can be developed to build a 4D spatio-temporal flower development model, which can be used for estimation of flower growth and visualization of flower development process.

Acknowledgement This work is supported in part by US National Science Foundation grants IIS-1231545 and IIS-1208420.

References

- [1] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194. ACM Press/Addison-Wesley Publishing Co., 1999.
- [2] D. Bradley, D. Nowrouzezahrai, and P. Beardsley. Image-based reconstruction and synthesis of dense foliage. *ACM Transactions on Graphics (TOG)*, 32(4):74, 2013.
- [3] T. Brox and J. Weickert. Level set segmentation with multiple regions. *Image Processing, IEEE Transactions on*, 15(10):3213–3218, 2006.
- [4] D. R. Fowler, P. Prusinkiewicz, and J. Battjes. A collision-based model of spiral phyllotaxis. In *ACM SIGGRAPH Computer Graphics*, volume 26, pages 361–368. ACM, 1992.
- [5] M. Fuhrer, H. W. Jensen, and P. Prusinkiewicz. Modeling hairy plants. *Graphical models*, 68(4):333–342, 2006.
- [6] T. Ijiri, S. Owada, M. Okabe, and T. Igarashi. Floral diagrams and inflorescences: interactive flower modeling using botanical structural constraints. *ACM Transactions on Graphics (TOG)*, 24(3):720–726, 2005.
- [7] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988.
- [8] C. Li, C. Xu, C. Gui, and M. D. Fox. Distance regularized level set evolution and its application to image segmentation. *Image Processing, IEEE Transactions on*, 19(12):3243–3254, 2010.
- [9] A. Lindenmayer. Mathematical models for cellular interactions in development i. filaments with one-sided inputs. *Journal of theoretical biology*, 18(3):280–299, 1968.
- [10] Y. Livny, F. Yan, M. Olson, B. Chen, H. Zhang, and J. El-Sana. Automatic reconstruction of tree skeletal structures from point clouds. In *ACM Transactions on Graphics (TOG)*, volume 29, page 151. ACM, 2010.
- [11] Y. Livny, S. Pirk, Z. Chengl, F. Yanl, O. Deussen, D. Cohen-Or, and B. Chenl. Texture-lobes for tree modelling. 2011.
- [12] L. Mündermann, P. MacMurchy, J. Pivovarov, and P. Prusinkiewicz. Modeling lobed leaves. In *Computer Graphics International, 2003. Proceedings*, pages 60–65. IEEE, 2003.
- [13] A. Myronenko and X. Song. Point set registration: Coherent point drift. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(12):2262–2275, 2010.
- [14] B. Neubert, T. Franken, and O. Deussen. Approximate image-based tree-modeling using particle flows. In *ACM Transactions on Graphics (TOG)*, volume 26, page 88. ACM, 2007.
- [15] M.-E. Nilsback and A. Zisserman. *An Automatic Visual Flora: Segmentation and Classification of Flower Images*. PhD thesis, Oxford University, 2009.
- [16] J. L. Power, A. Brush, P. Prusinkiewicz, and D. H. Salesin. Interactive arrangement of botanical l-system models. In *Proceedings of the 1999 symposium on Interactive 3D graphics*, pages 175–182. ACM, 1999.
- [17] P. Prusinkiewicz. Graphical applications of l-systems. In *Proceedings of graphics interface*, volume 86, pages 247–253, 1986.
- [18] P. Prusinkiewicz, M. S. Hammel, and E. Mjolsness. Animation of plant development. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 351–360. ACM, 1993.
- [19] P. Prusinkiewicz, L. Mündermann, R. Karwowski, and B. Lane. The use of positional information in the modeling of plants. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 289–300. ACM, 2001.
- [20] L. Quan, P. Tan, G. Zeng, L. Yuan, J. Wang, and S. B. Kang. Image-based plant modeling. In *ACM Transactions on Graphics (TOG)*, volume 25, pages 599–604. ACM, 2006.
- [21] A. Runions, M. Fuhrer, B. Lane, P. Federl, A.-G. Rolland-Lagan, and P. Prusinkiewicz. Modeling and visualization of leaf venation patterns. In *ACM Transactions on Graphics (TOG)*, volume 24, pages 702–711. ACM, 2005.
- [22] P. Tan, G. Zeng, J. Wang, S. B. Kang, and L. Quan. Image-based tree modeling. *ACM Transactions on Graphics (TOG)*, 26(3):87, 2007.