Analyzing Growing Plants from 4D Point Cloud Data

Yangyan Li[1] Xiaochen Fan[1] Niloy J. Mitra[2] Daniel Chamovitz[3] Daniel Cohen-Or[3] Baoquan Chen[1,4,*]
[1]Shenzhen,VisuCA Key Lab/SIAT [2]University College London [3]Tel Aviv University [4]Shandong University

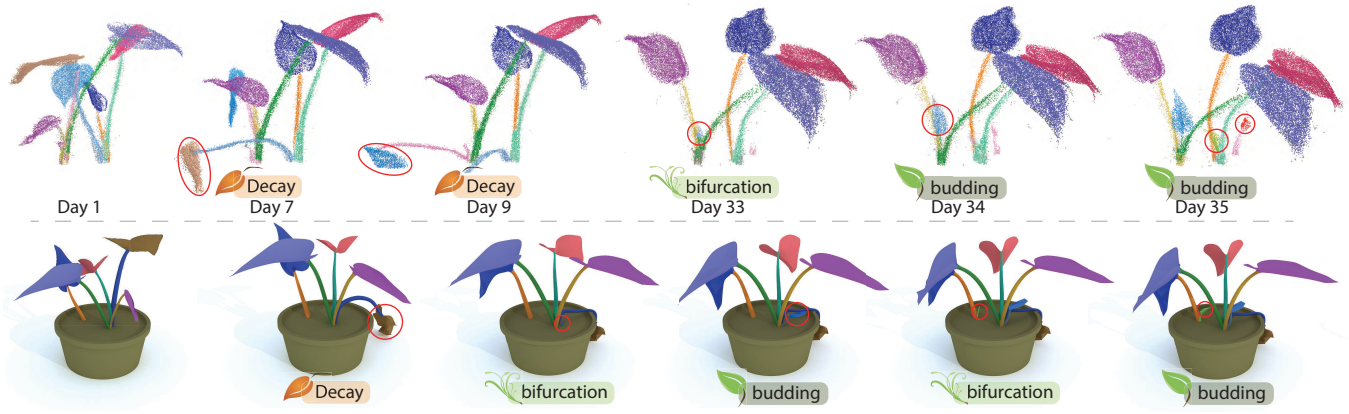

Figure 1: *(Top) Dishlia growth time lapse point cloud over 5 weeks, with classified organs and detected budding, bifurcation and decay events. (Bottom) The extracted events are then used to bring a static plant model to life with both motion and growth.*

Abstract

Studying growth and development of plants is of central importance in botany. Current quantitative are either limited to tedious and sparse manual measurements, or coarse image-based 2D measurements. Availability of cheap and portable 3D acquisition devices has the potential to automate this process and easily provide scientists with volumes of accurate data, at a scale much beyond the realms of existing methods. However, during their development, plants grow new parts (e.g., vegetative buds) and bifurcate to different components — violating the central incompressibility assumption made by existing acquisition algorithms, which makes these algorithms unsuited for analyzing growth. We introduce a framework to study plant growth, particularly focusing on accurate localization and tracking topological events like budding and bifurcation. This is achieved by a novel forward-backward analysis, wherein we track robustly detected plant components back in time to ensure correct spatio-temporal event detection using a locally adapting threshold. We evaluate our approach on several groups of time lapse scans, often ranging from days to weeks, on a diverse set of plant species and use the results to animate static virtual plants or directly attach them to physical simulators.

Keywords: growth analysis, 4D point cloud, event detection

Links: DL PDF WEB VIDEO DATA

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric Algorithms; I.4.6 [Image Proc. and Computer Vision]: Segmentation—Point Classification;

1 Introduction

Studying growth processes in organic life forms has a long history in science. Traditionally, such studies rely on manual recordings of growth stages, or image-based measurements taken at sparse intervals. Such workflows are tedious, prone to measurement bias, and difficult to scale to large-scale observations, both in space and time. Advances in affordable 3D acquisition devices provide new opportunities.

Plant growth is fundamentally different from animal growth. Most animals are born with all their body organs, which grow and mature with age. In contrast, plants grow and develop throughout their life cycle, constantly producing new tissues and structures [Chen and Laux 2012]. Studying such developments involves detecting specific growth events (e.g., budding of a leaf, see Figure 2), quantifying these events, and tracking their subsequent evolution over time. Beyond difficulties arising from motion, the key challenge is to track the continuous shape changes in geometry and topology, albeit at a very slow rate, due to growth, and possibly due to decay. This is different from typical motion capture setups studying human movements.

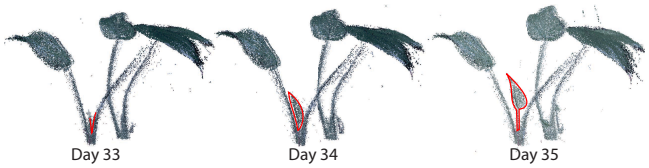

Figure 2: *Challenges in detecting bifurcation/budding events from point cloud. We observe that the bifurcation on day 33 and the budding on day 34 are subtle and hard to identify; the organs mature and become easier to detect later on day 35.*

Recent research in acquisition methods has led to rapid advances in recovering dynamic geometry acquisition while accounting for missing data, outliers, and movements (e.g., rigid, articulated, non-rigid) among multiple scans across time. All these methods, however, assume the underlying object to be *incompressible*, i.e., objects can deform but not grow (or decay). Thus, tracking plant growth (or any growth in a larger context) requires fundamentally different data analysis algorithms.

We propose an interleaved spatial and temporal analysis in 4D, where the challenge is to accurately locate budding and bifurcation events. To that end, we present a *forward-backward* analysis, where moving forward in time we identify the 'after-affect', i.e., we extract an easier to detect future event and pull it back in time to accurately locate the event while still in its infancy (see Sections 4 and 5). We show the necessity of this forward-backward process, and its effectiveness in accurately locating (morphological) events leading to novel automated quantitative measurement and analysis for studying plant growth (see Figure 2).

We validated our method on synthetic datasets and successfully detected growth (and decay) events in scan sequences of a variety of plant species (e.g., Anthurium, Dishlia, etc.) often acquired over a period of days and weeks (Section 6). Beyond implications in studying the underlying growth laws of plants, the recovered growth parameters are immediately useful for animating plant growth and simulating, which are otherwise very tedious work for artists (Section 7). We use the evolving organs consistently segmented out from the 4D point cloud to synthesize live plant models with both growth and motion. The organ information as well as the properties associated with each organ can be fed into plant simulators to produce simulations that accurately mimic observed motions in reality.

2 Related Work

3D and 4D reconstruction. Raw output of 3D scanners suffers from various imperfections including noise, missing data, and outliers. Different methods have been developed to improve the quality of the corresponding reconstructed models (e.g., [Curless 1999; Alexa et al. 2001; Kazhdan et al. 2006; Huang et al. 2009] and references therein). Such methods mostly factor out camera movements and consolidate the scans across different viewpoints while assuming the scanned object to be static.

When, in addition to camera motion, objects move or deform, time information becomes critical. Hence, 4D reconstruction with time being the fourth dimension is used for accurate capture of human motion or facial expressions while ensuring spatio-temporal coherence across frames. A common approach is to fit pre-defined shape templates, which encode the topology and sometimes also the coarse geometry of the captured shape, while solving for per-frame pose variations [Ahmed et al. 2008; de Aguiar et al. 2008; Vlasic et al. 2008; Bradley et al. 2008; Pons-Moll et al. 2011]. When successive scans have small relative deformation and share large overlapping regions, adequate feature correspondences can be found between consecutive frames. In such cases, shapes and deformations can be directly recovered without the need for any prior template priors [Mitra et al. 2007; Liao et al. 2009; Wand et al. 2009; Popa et al. 2010; Tevs et al. 2012; Akhter et al. 2012]. In the case of video sequences, Beeler et al. [2011] find predefined anchor frames and propagate the information both forward and backward to establish frame correspondence. In addition to deforming, plants also *grow* (both discrete and continuous) over time and thus violates a key assumption in all the above methods.

*Corresponding author.

Incompressible models and fixed topology. In many applications, the scanned objects preserve volumes (i.e., incompressible) even under deformations and articulation. Hence, Sharf et al. [2008] use an incompressibility assumption to perform volumetric reconstruction by modeling the material flow. Similarly, consistency in topology can be used to regularize the registration, such as reduced deformable models in Chang et al. [2009; 2011], consensus skeleton across frames in Zheng et al. [2010], or temporally coherent hole filling in Li et al. [2012].

While shape templates, incompressibility, and topology consistency priors are valid for many objects, and hence effective for their reconstructions, they often do not apply to plants. Specifically, plants evolve continuously as organs emerge, build up and wither, resulting in significant changes in shape (e.g., leaf curling), volume (e.g., leaf expansion and bud formation), and topology (e.g., stem bifurcation). Recently, Bojsen-Hansen et al. [2012] presented a method to recover a temporally coherent, deforming triangle mesh with arbitrarily changing topology from an incoherent sequence of static closed mesh surfaces. In the medical imaging setting, Lu et al. [2012] estimate the probability of detected tumors in MRI scans using non-rigid registration in a Bayesian setting. We not only focus on deforming shapes but also on changing topologies as necessary to accurately capture growth-related activities.

Plant modeling and simulation. Procedural generation of plants has received much attention in computer graphics ([Rozenberg and Salomaa 1980; Prusinkiewicz and Lindenmayer 1996]). While it is possible to create very realistic looking plants using L-systems, we are interested in capturing both form and dynamics of real plant growth. The analyzed data can be in turn used to re-create high quality geometry or 'animate' procedural plants accurately and realistically, which is very tedious to achieve manually. A variety of methods have been developed to generate static plant models from different data sources [Quan et al. 2006; Xu et al. 2007; Neubert et al. 2007; Livny et al. 2010]. Mündermann et al. [2005] presented an approach to recover a descriptive developmental model for Arabidopsis with a large amount of measurement data from sparsely time-lapse images. Fernandez et al. [2010] proposed a semi-manual approach to track plant growth at cell resolution from 4D confocal data, focusing on segmenting cells of each frame and computing the cell lineages between frames. We refer the reader to the comprehensive survey of [Prusinkiewicz and Runions 2012].

Our work is more related to the recent work of Li et al. [2011], who model and generate moving trees from video, Pirk et al. [2012b], who leverage dynamic tree modeling and representation for adapting complex tree models to their environment interactively, and Pirk et al. [2012a], who compute developmental stages of a tree. The methods approximate trees' natural growth following botanic growth models and allometric rules. Although we target a similar goal, we capture live plants and directly track their growth without assuming access to an underlying growth template or bifurcation rules.

Recently, Zhao and Barbič [2013] proposed a technique to interactively author plant models to be simulation-ready by decomposing the polygon soup into domains and building a hierarchy between them. While it is very challenging and often ambiguous to automatically segment plants into individual organs from isolated plant models, we demonstrate that such segmentation can be robustly extracted from a raw 4D point cloud, making the resultant models directly useable by their simulation framework, and thus providing a complementary way to generate simulation-ready models.

Event detection. Apart from reconstruction, time lapse of 2D images or 3D point clouds contain valuable data that can be ana-

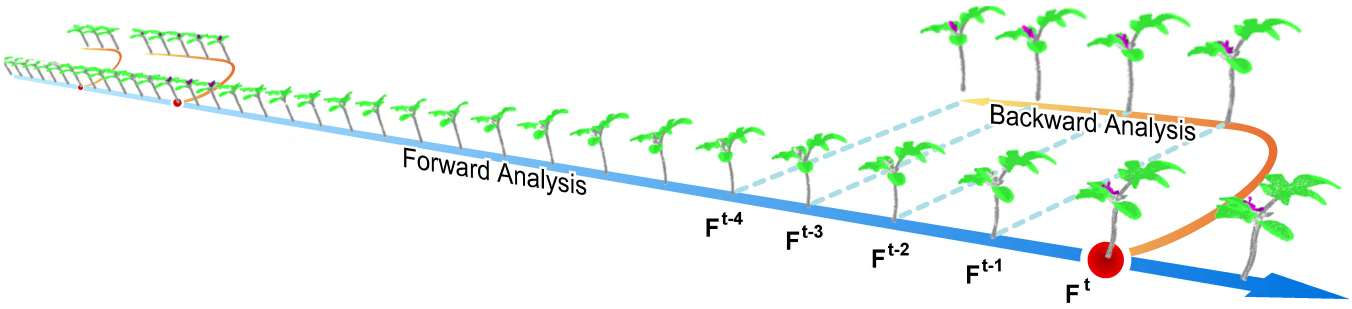

Figure 3: *The forward and backward analysis. The forward analysis chronologically processes the frames seeking for the strong evidences of events, in which cases the backward analysis is triggered to refine the detected spatio-temporal events. In this figure, a new leaf is detected at F^t (highlighted in purple) by the forward analysis, then the backward analysis is triggered, carrying the information from the future, to refine the events both in space and time.*

lyzed to detect and understand particular events that occurred during the captured time periods. Brendel et al. [2011] and Gaur et al. [2011] convert training videos into spatio-temporal preserving representations, to allow the detection and localization of relevant activities by matching them to the training data. Pirsiavash et al. [2012] presented a first-person view camera involving long-scale temporal structure and complex object interactions as well as analysis on detecting activities in the data set. Kevin et al. [2012] utilize a conditional model that is able to automatically discover discriminative and interesting segments of video. Shotton et al. [2011] recognize human pose as an instance of per-pixel classification problem towards a real-time solution. Instead of a supervised approach, we directly detect budding events by analyzing the morphology of the inspected plant, without access to any training set. Kalal et al. [2010] measure tracking quality based on a *forward-backward error* instead of errors only from neighboring frames, thus achieving robustness against local ambiguities. We also propose a forward-backward analysis but for accurately detecting and tracking events, which are often subtle to automatically identify in their early stages.

Thus, to the best of our knowledge, we present the first attempt focusing on event detection in raw point clouds, and analyzing growth patterns, both in geometry and changing topology across time, in the context of growing (and decaying) plants.

3 Acquisition Setup

Our acquisition system, shown in Figure 4, consists of a standard structured light scanner with a camera-projector pair [Song and Chung 2008; Song et al. 2013] and a turntable on which to place the plant. We have an array of four such systems, each of which can independently capture and record the growth of a plant. Each setup is housed in its own glass cage to shield the plants from external disturbances (e.g., wind).

The scanner captures the geometry of the plant from a specific view, while the turntable rotates by 30 degrees each time. After each turn, the setup pauses for 5 sec for the plant to stabilize before scanning the plant. A full cycle scan consists of 12 such single-view scans. Intermediate growth and subtle movements during this multi-view scanning are ignored. We factor out the known movements of the turntable to bring the scans to a consistent coordinate system and refine this initial alignment using a multiview ICP refinement. In each view, the camera records 30 shots over 3 seconds. It takes 5 minutes to do a full cycle scan, which we call a *frame* and denote by F^t (the superscript t indicates the time instant). The system records 12 frames per hour for an average of 25 days.

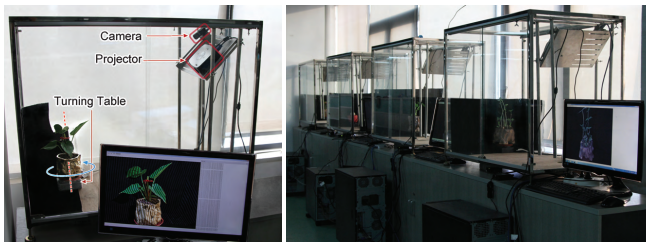

Figure 4: *(Left) Our acquisition setup consists of a series of structured light scanners. (Right) Plants are housed in individual glass cages to ensure controlled growth and stable environments.*

The resolution of the structured light stripe images is 1280×960 and the camera is about 1 m away from the plant, resulting in an error range for the generated points to be less than 1 mm. However, in our setting, measurement errors were a bit higher since we provide sufficient light access to the plants, which is less than ideal for the structured light device.

We captured massive volumes of structured light images. The size of the resultant point cloud directly depends on the coverage of the plant in image space, typically ranging between 10-500K, summing up to 3-150M points per day (excluding points from the flower pots). We found the turntable to be unstable over a long time period, so we use the features on the flowerpots together with the turntable axis to refine the registration. We first apply pairwise ICP to the flowerpot points of neighboring views and then use the constraint that the 12 views are coaxial in the joint GraphSLAM [Thrun and Montemerlo 2005] to register the points across the different views. The resultant registration is used to update the axis of the turn table for the following frames (data freely available for academic use).

While we use known technology in the core steps, the main challenge was practical – how do we build a very robust acquisition setup that can scan over long durations (spanning a few weeks) without significant drift or alignment issues? More advanced 3D acquisition (e.g., Coplanar Shadowgrams [Yamazaki et al. 2007; Yamazaki et al. 2009]) specifically designed for long time scanning can alternately be used in this stage.

4 Forward-Backward Analysis

Starting from input 4D point cloud data that records development of a plant, our main goal is to track growth of the plant. In particular, we are interested in segmenting the plant, tracking the growth

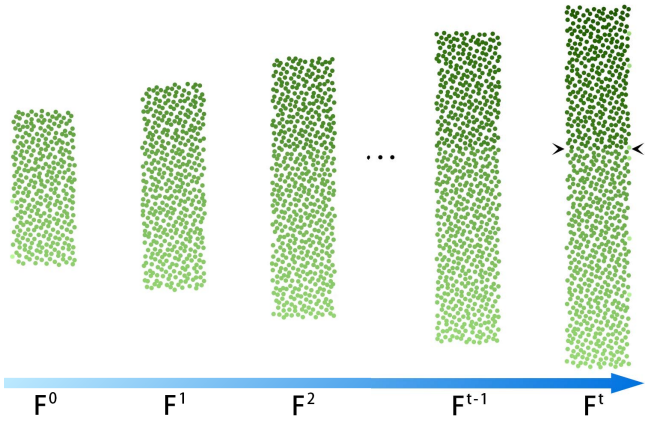

Figure 5: *Bar 'Budding'. Labeling the two vertical bars become easy at the right end, when they are mature. Labeling them on left end, close to the budding event, is extremely difficult. However, incrementally tracking from the right backward to the left, and considering the known labeling of an adjacent frame, significantly improves the accuracy of the labeling at the left end.*

of its organs, e.g., stems and leaves, and identifying events such as vegetative *budding* of a leaf, or *bifurcation* of a stem. In other words, we would like to associate each organ with a unique label, and detect the spatio-temporal event of the organ's birth or death (i.e., budding, bifurcation, or decay).

Unlike typical acquisition settings where the challenges arise from moving camera, articulated, or deformable body movements, or occluded parts, our problem is different. We have to account for emerging object parts since our key focus is to detect such growth events, i.e., locate them both in space and time. There are two key challenges: (i) events, when they first occur, are too subtle to accurately locate, and only later, after a few frames, can they be robustly detected; and (ii) looking too far into the future is also misleading as events can meanwhile disappear (e.g., decay of a leaf), or the organs can become occluded by other ones. We propose a mixed approach: step *forward* through frames, detecting strong evidence that an event occurred in the past, and then step *backwards* to correctly locate the origin of the spatio-temporal event (see Figure 3).

Growth events leave noticeable traces in the near future as the bifurcated stem or the grown bud. Both of these organs are relatively easy to detect in a few frames *after* the actual event by analyzing the point cloud using local discriminating features. In particular, curvature helps discriminate between leaves and stems; while local orientation helps discriminate different branches of the stem around bifurcation points.

Since growing parts gradually evolve over time, feature differences are more prominent when measured with respect to a local *reference* value observed in nearby frames, instead of with respect to absolute values. Hence, the classification parameters are better derived directly from the local data, rather than being hard-coded at the beginning. This implies that the reference value must be retrieved from the future, where the reference value exits for the organ for which we search its budding, and no reference values exist in the past. This again suggests that we can only go forward to the future to have an effective reference value to analyze the data, before stepping back in time.

Figure 5 illustrates a vertical slab over which a brand new slab emerges. The two adjacent slabs grow vertically with the horizontal axis denoting time. The slabs are colored by their feature values. After some time their features clearly indicate the existence of two slabs, while on the left end, say in F^0, or in F^1, when the new upper slab is budding, their features are quite alike and very difficult to discriminate. However, if the features associated with the two slabs in frame F^1 are known, they can be used as local references for defining effective relative features, allowing discriminating between the two slabs in F^0. Thus, once we detect a clear split between the two slabs, say in frame F^t, we define the two local reference values, and trace from right to left (i.e., backwards in time), until one slab vanishes at $F^{t-\delta}$. Then the *new slab* event is detected at $F^{t-\delta+1}$.

One of the advantages of our forward and backward approach is that it can be applied on-the-fly during the acquisition of the data. While the hardware captures F^{t+1}, the algorithm has enough time to analyze from frame F^t, and if needed, tracing backwards to detect the event in the short past. Note that, for decay events, the strong evidences come before they become subtle, and can thus be robustly detected in the forward analysis.

5 Growth Event Detection

Detecting events by counting organ number. Growth events are qualitative changes resulting in the increase or decrease of organ number between neighboring frames, e.g., budding and bifurcation events increase leaf and stem numbers, while decay events decrease them. Critical to event detection are the leaf number and stem number in each frame, and can be compute by decomposing point cloud of each frame into individual organs. We then check the organ numbers between adjacent frames to detect growth events.

Solving organ decomposition as a labeling problem. We compute the organ number of each frame by first generating a organ hypothesis of one frame, with organ number no less than the real one, then validating the hypothesis and thus narrowing the hypothesis number down to the real one. The organ hypothesis of each frame is generated by analyzing the geometry of that frame, as well as referencing to the adjacent frames. We then map the points to the organs by minimizing suitable energy functions. Organs without any assigned points indicate false positives, and hence are removed. The mapping from the points to the remaining organs decomposes the point cloud into individual organs and thus provides a count of organs for detecting growth events.

Specifically, we maintain an *active organ hypothesis* $\mathcal{H}^t := \{L_l^t\} \cup \{S_s^t\}$ for frame F^t, where L and S denote leaf and stem categories, with L_l^t as the l-th leaf and S_s^t as the s-th stem, respectively. Given a point $p^t \in \mathcal{P}^t$ in F^t, we would like to find a labeling that maps $\mathcal{P}^t$ to $\mathcal{H}^t$, while minimizing a suitable energy function. Such labeling outputs both the organ category (L or S) and the index (l^t or s^t) of the organ p^t should belong to.

Two-stage labeling. Instead of simultaneously solving for organ category and organ indices, we take a two-stage approach. The first stage finds a binary labeling f_B that maps $\mathcal{P}^t$ to $\{L, S\}$ and thus classifies the point cloud into leaf points L^t and stem points S^t; the second stage consists of two multi-labelings f_L and f_S, further decomposing L^t and S^t into individual leaves $\{L_l^t\}$ and stems $\{S_s^t\}$, respectively.

Both the binary and multiple labeling stages in both forward and backward analysis make use of information from adjacent frames to generate and validate the active organ hypothesis $\mathcal{H}^t$. In the multiple labeling stage, the labels, and even the number of them, are unknown. The organ information from adjacent frames is especially useful for the multiple labeling stage to automatically gener-

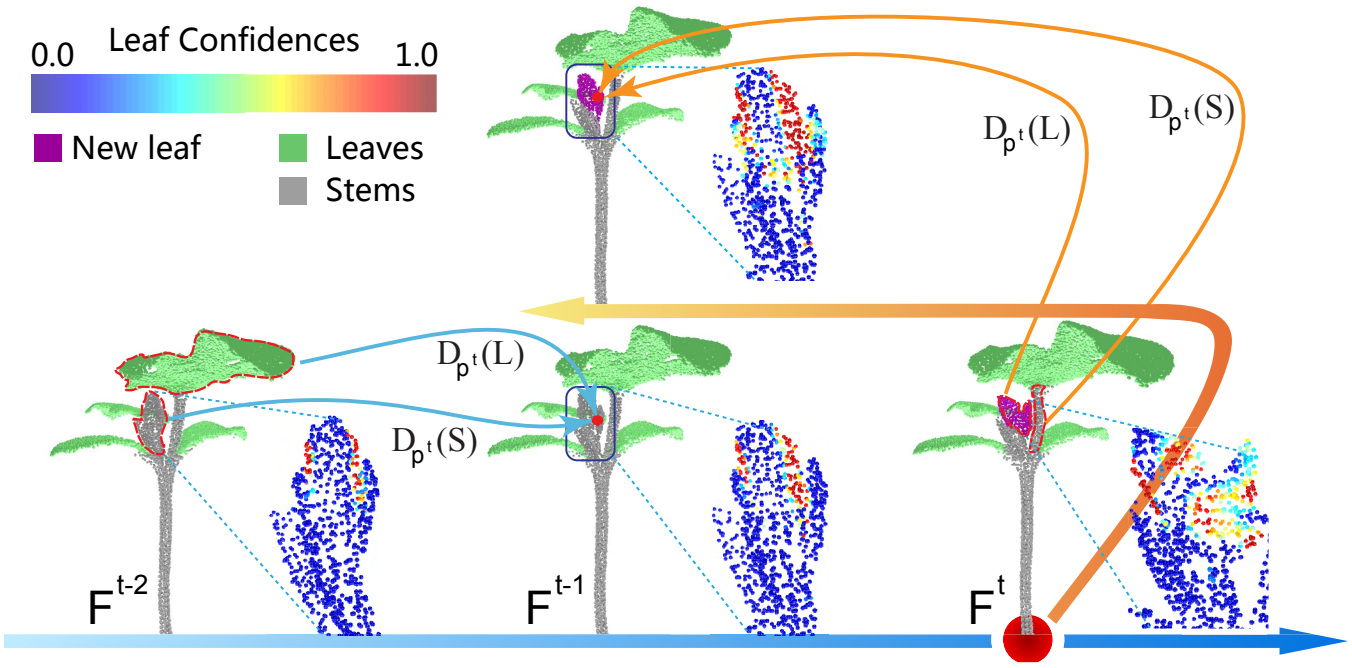

Figure 6: *Feature reference adaptation. A zoom of F^{t-2} to F^t in Figure 3 with associated feature confidences. The confidences for forward analysis are computed against the relative features that come from the past, while the confidence for the backward analysis are computed from the relative features coming from future. We show that the relative features gives higher confidences leading to better localization of events in time and space. The illustration demonstrates the necessity of adapting thresholds to handle the development of plant organs.*

ate the target labels for the the points of current frame to be mapped to. Depending on the forward/backward direction, $\mathcal{H}^{t-1}$(in forward analysis) or $\mathcal{H}^{t+1}$(in backward analysis) will be taken as reference while processing F^t. We next discuss the two stages in detail.

5.1 Leaf-stem binary classification

We find a binary labeling f_B that maps $\mathcal{P}^t$ to two known labels L and S for leaf and stem category, respectively. We want each point to be mapped to the most likely organ type, while neighboring points should be mapped to the same organ type for spatial coherence. We accordingly define the objective energy to be minimized by the labeling as:

$$E(f_B) := \sum_{p^t \in \mathcal{P}^t} D_{p^t}\big(f_B(p^t)\big) + \sum_{(p^t,q^t)\in\mathcal{N}_{\mathcal{P}^t}} V\big(f_B(p^t), f_B(q^t)\big), \quad (1)$$

where $\mathcal{N}_{\mathcal{P}^t} := \{(p^t, q^t)\}$ is a neighborhood system on $\mathcal{P}^t$ with $\{(p^t, q^t)\}$ being 3D Delaunay triangulation edges with Euclidean lengths less than 3mm (according to the error range of our setup). The data term $D_{p^t}\big(f_B(p^t)\big)$ measures the cost of classifying p^t as leaf or stem. The smoothness term $V\big(f_B(p^t), f_B(q^t)\big)$ measures the cost of assigning the labels $f_B(p^t)$, $f_B(q^t)$ to the neighboring points p^t, q^t and is used to encourage spatial coherence. We approximately solve the minimization via an Markov Random Fields (MRF) formulation using α-expansion [Boykov et al. 2001; Boykov and Kolmogorov 2004; Kolmogorov and Zabih 2004].

At the borders of organs, neighboring points often have very different labels and it is important that the energy function do not over penalize such labeling. Since curvature estimates are typically high around organ boundaries, we define the smoothness term as:

$$V\big(f(p^t), f(q^t)\big) := \begin{cases} \max(\frac{1}{C(p^t)}, \frac{1}{C(q^t)}) & \text{if } f(p^t) \neq f(q^t) \\ 0 & \text{if } f(p^t) = f(q^t) \end{cases}, \quad (2)$$

where $C(p^t)$ is the curvature of p^t, and the labeling function f applies for f_B here, as well as f_L and f_S in Section 5.2. The smoothness term penalizes labeling neighboring points to different organs, but less near organ borders. The curvature value is approximated as $C(p^t) := \lambda_1/(\lambda_1 + \lambda_2 + \lambda_3)$, with $\lambda_1 \leq \lambda_2 \wedge \lambda_1 \leq \lambda_3$, using the eigenvalues from a local PCA analysis of points around p^t.

A mature leaf is typically flatter compared to stems, and hence has points with lower curvature values. We use this observation to discriminate between stems and mature leaves. Hence, we build the data term $D_{p^t}\big(f_B(p^t)\big)$ based on the curvature values.

Since $C(p^t) \in [0, 1/3]$, larger values can be several times the smaller ones, we first cap very small $C(p^t)$ to c_ϵ (we use 0.015), and then use a log function to further reduce the non-linearity of the feature values, resulting in flatness feature $R(p^t) := \log\big(\max(C(p^t), c_\epsilon)\big)$.

We define $\Re_L := \log(c_\epsilon)$ and $\Re_S := \log(1/3)$, and $R(p^t) \in [\Re_L, \Re_S]$. With the flatness feature of each point $R(p^t)$, we define flatness feature of each leaf $R(L_l^t) := \sum_{p^t \in L_l^t} R(p^t)/|L_l^t|$ and stem $R(S_s^t) := \sum_{p^t \in S_s^t} R(p^t)/|S_s^t|$, respectively.

We define the distance between a point p and an organ O, $d(p, O)$, as the distance between p and the closest point in organ O. We take the flatness feature $R(L_{l*}^{t\pm1})$ of the closest leaf $L_{l*}^{t\pm1}$ and flatness feature $R(S_{s*}^{t\pm1})$ of the closest stem $S_{s*}^{t\pm1}$ in F^{t-1}(in forward analysis) or F^{t+1}(in backward analysis) as the reference value for computing the data cost of labeling p^t to leaf or stem (see Figure 6), and only when the closest leaf or stem is missing in the neighboring frames, global absolute threshold $\Re_L$ and $\Re_S$ are used:

$$D_{p^t}(L) := \begin{cases} \max\big(R(p^t) - R(L_{l*}^{t\pm1}), 0\big) & \text{if } \Phi > 0 \\ R(p^t) - \Re_L & \text{if } \Phi = 0 \end{cases}, \quad (3a)$$

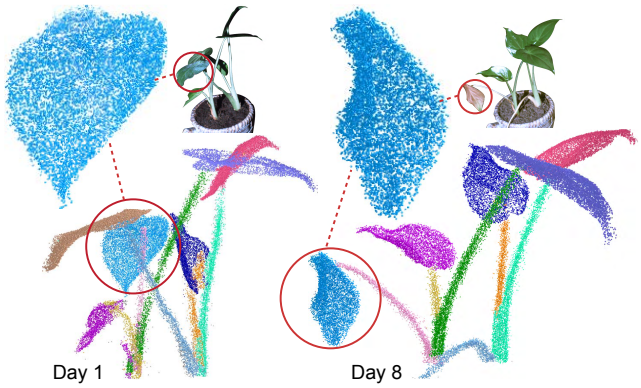

Figure 7: *A decaying Dishlia leaf (circled). As can be seen from the enlarged rendering, the leaf shrinks and folds as it decays, yet it can be correctly classified through its degrading process.*

$$D_{p^t}(S) := \begin{cases} \max\left(R(S_{s*}^{t\pm1}) - R(p^t), 0\right) & \text{if } \Phi > 0 \\ \Re_S - R(p^t) & \text{if } \Phi = 0 \end{cases}, \quad (3b)$$

where $\Phi := |\{L_l^{t\pm1}\}| \times |\{S_s^{t\pm1}\}|$ is an indicator of whether there are both leaf and stem existing in the neighboring frame. The relative definition of the data term adapts both in time (neighboring frame) and space (closest organ), and thus accommodates for gradual evolution of plant parts.

The forward-backward analysis (see Section 4) with the adaptive reference values in the data term enables accurate leaf-stem classification even when the differences are subtle or when the flatness feature values are biased, while forward only analysis with global absolute threshold misses them in their early days (see Table 1).

5.2 Individual organ segmentation

The leaf-stem binary classification decomposes $\mathcal{P}^t$ into leaf points L^t and stem points S^t, which are further decomposed by multi-labelings f_L and f_S into individual organs. Labelings f_L and f_S can be performed in parallel, since they operate on different parts of the data. We elaborate first f_L and then f_S in this subsection.

Leaf multi-labeling segmentation. The leaf-stem binary classification already takes the stem points S^t away from $\mathcal{P}^t$, leaving the leaf points, without the support of stems, floating in the air. Ideally, the connected components in L^t should represent individual leaves. But in cases that two leaves touch or partially overlap, they will be treated as one segmentation. To address this issue, we employ temporal coherence, i.e., using the segmentation results of the adjacent frames to resolve ambiguity in the current frame.

We generate leaf hypothesis $\{\Omega_l^t\}$ by jointing the potential leaves $\{L_l^t\}$ in F^t with leaves detected in F^{t-1}(in forward analysis) or F^{t+1}(in backward analysis): $\{\Omega_l^t\} \Leftarrow \{L_l^t\} \uplus \{L_l^{t\pm1}\}$. The intuition of the $\uplus$ operator is that if two leaves were previously separated and correctly segmented, this information should be transferred to the current frame when the two are now connected. Note that the $\uplus$ operator may put some false positives in the hypothesis. We segment the leaf points into individual leaves by solving a multi-labeling f_L which maps $p^t \in L^t$ to the joint leaf hypothesis $\{\Omega_l^t\}$, when the following energy function $E(f_L)$ is minimized:

$$E(f_L) := \sum_{p^t \in L^t} D_{p^t}(f_L(p^t)) + \sum_{(p^t,q^t)\in\mathcal{N}_{L^t}} V(f_L(p^t), f_L(q^t)), \quad (4)$$

where $D_{p^t}\big(f_L(p^t)\big) := d\big(p^t, f_L(p^t)\big)$, and the neighborhood system is on L^t, instead of $\mathcal{P}^t$. After the optimization, there may be empty leaves, which are the false positives in the hypothesis. They are removed from $\{\Omega_l^t\}$ and the remaining are used to update $\{L_l^t\} \Leftarrow \{L_x \in \{\Omega_l^t\} : L_x \neq \emptyset\}$ (see Figures 10 and 11).

Once the leaves of F^t are computed/updated, their flatness features are computed/updated accordingly. The updated values are used for adapting reference feature values in the process of the adjacent frames. Figure 6 shows an example for detecting evolving or degrading organs. Note that the reference values are processed from strong growth event evidences to subtle cases, thus the evidences of bifurcation or budding events are detected in the forward analysis and then refined in the backward analysis with the information from future. The decaying events come with strong evidences with subtle endings, thus can be robustly handled in the forward analysis (see Figure 7).

Stem multi-labeling segmentation. Like that in the leaf segmentation part, the stem segmentation decomposes stem points S^t into individual stems $\{S_s^t\}$. However, unlike leaves that are largely isolated by the leaf-stem classification, the stems points are still tightly connected.

Instead of extracting connected components, we *branch* the stem points into potential stems (c.f., [Huang et al. 2013]). Here we have a threshold ℓ for trimming the short stems, but it performs similar as $\Re_L$ and $\Re_S$ for leaf-stem classification: although we take a conservative decision, branches could be mistakenly trimmed, still, the backward analysis will have a chance to fix this. Finally, we get a set of individual branches (see Figure 8).

Similar to leaf segmentation, we generate stem hypothesis by $\{\Omega_s^t\} \Leftarrow \{S_s^t\} \uplus \{S_s^{t\pm1}\}$, and then find the labeling f_S that maps $p^t \in S^t$ to $\{\Omega_s^t\}$ by minimizing:

$$E(f_S) := \sum_{p^t \in S^t} D_{p^t}(f_S(p^t)) + \sum_{(p^t,q^t)\in\mathcal{N}_{S^t}} V(f_S(p^t), f_S(q^t)). \quad (5)$$

The orientation feature are discriminative between different branches, upon which the data term is built:

$$D_{p^t}(f_S(p^t)) := \begin{cases} O(p^t) - O(f_S(p^t)) & \text{if } d(p^t, f_S(p^t)) \le \Delta \\ \Upsilon & \text{if } d(p^t, f_S(p^t)) \ge \Delta \end{cases}, \quad (6)$$

where $O(p^t) := (x, y, z)$ is the orientation of p^t, with x, y, z the diagonal element of the matrix from the local PCA analysis, $O\big(f_S(p^t)\big)$ is the average orientation of points in $f_S(p^t)$, and Υ is a large penalty that prevents p^t being assigned to a distant stem. After the optimization, the stem hypothesis is updated: $\{S_s^t\} \Leftarrow \{S_x \in \{\Omega_s^t\} : S_x \neq \emptyset\}$.

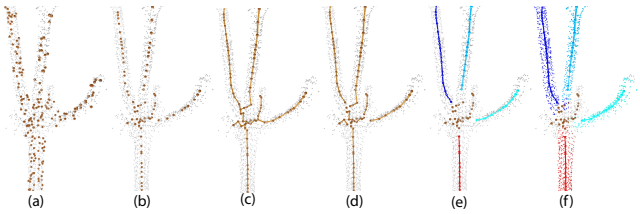

Figure 8: *Branch extraction from the stem points:* $1/10$ *points are sampled from the stem points (a), these samples are centralized and further downsampled (b), then a minimum spanning tree is applied to generate the compound stem (c), then the MST is filtered by node degree (d). The remaining branches that are shorter than a threshold ℓ are trimmed (e). Finally, branches are mapped to the stems points and stem components are extracted (f).*

Table 1: *Dataset. The first two columns after the plant names describe the length of scanning (in days) and point density (number of points per frame).* Initial *and* Final *describe the number of leaves and stem segments at the start and end of our study, respectively. The* Events(f&b) *provides a chronically record of events such as budding, bifurcation, and decaying detected by our algorithm. The* Events(f-only) *provides events detected by forward only analysis with global absolute threshold.*

Set	#day	#pts	Initial	Final	Events(f&b)	Events(f-only)
Anthurium	18	580k	#stem: 10 #leaf: 5	#stem: 10 #leaf: 8	day 3: bud. day 3: bud. day 12: bud.	day 4: bud. day 5: bud. day 14: bud.
Dishlia	35	1,060k	#stem: 6 #leaf: 6	#stem: 7 #leaf: 7	day 7: dec. day 9: dec. day 17: bif. day 27: dec. day 29: bif. day 33: bif. day 33: bif. day 34: bud. day 35: bud. day 35: bud.	day 6: dec. day 8: dec. day 19: bif. day 24: dec. day 32: bif. day 34: bif. day 35: bif. day 35: bud. day 35: bud. day 35: bud.
Dancing bean	5	500k	#stem: 3 #leaf: 0	#stem: 5 #leaf: 2	day 1: bif. day 2: bif. day 5: bud. day 5: bud.	day 2: bif. day 2: bif. day 5: bud. day 5: bud.

Detection of new organs in the current frame of the forward analysis, triggers the backward analysis to update the leaf-stem classification, as well as individual organ segmentation of previous frames. Both the leaf and stem segmentation procedures work at a part-level and generate consistent segmentation across frames. The algorithm establishes organ correspondences between frames and separating touching/overlapping leaves (see (see Figures 10 and 11), which is essential for tracking individual organs and detecting new events.

6 Results

We have captured data representing the development of several plants over a period of 1-5 weeks. All the plants grow from seeds, but some are moved into the acquisition cube while the plant was already several inches tall, due to the special care needed at the early stage. After that, the plants remain in the cube until they grow out of sight of the camera, and we stopped the acquisition. We used three growing plants representing different species in this study (the number of weeks grown for are shown in parentheses): an Anthurium (3)-in Figures 11 and 9, a Dishlia (5)-in Figure 1, 10, and 7, and a dancing bean (1)-in Figure 12. Table 1 lists some statistics of these three plants.

Each plant represents different shapes and different growth dynamics: the Anthurium plant has large leaves with firm branches; the

Figure 9: *Anthurium growth time lapse point cloud (colored according to photos) over 18 days (top row), with classified organs (e.g., leaf and branch, distinctively colored) and detected development events shown in the bottom row.*

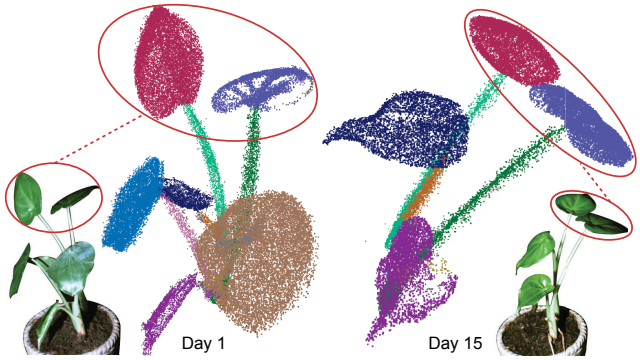

Figure 10: *Two frames of the Dishlia plant with organ segmentation. Two leaves grow into almost the same plane and touch each other in Day 15, which is difficult to segment only using this frame. However, they are distinguished by tracking forward from when they were spatially separated and thus correctly labeled in Day 1.*

Dilisha plant has relatively thinner branches, therefore, leaves have larger motion than the Anthurium plant; the Dancing Bean plant is the fastest growing, and because of its large swaying movements we refer to it as the "Dancing Bean" (see Figure 12).

As a plant matures, its structures grow dense, leading to significant occlusions and direct physical interactions among its various organs. This results in blurred or even indistinguishable boundaries between structures, which is made worse by the noisy nature of the input scans. At any frozen moment of time, it can be difficult to discern semantically meaningful plant components. Figure 11 shows a frame of the Anthurium data, where one leaf touches another leaf making them hard to distinguish from each other. However, several frames after, that leaf grows away from the other leaf and moves to open space. The leaf-stem classification for this frame immediately identifies the new leaf. This event in the forward analysis, triggers the backward analysis to track and detect the leaf closer to its budding moment. This supports the underlying assumption of temporary coherence during the course of the plant's growth. Figure 10 shows another example of forward analysis.

Beyond detecting growth, we also track decay (i.e., senescence) of leaves, when they fold or wither away. Figures 7 shows a decaying leaf of the Dishlia plant. Note that the geometric shapes of the same plant component can be significantly different. The temporal coherence helps to resolve the ambiguity. Figure 1 depicts more events detected in the Dishlia plant growth including budding, bifurcation, and leaf decay.

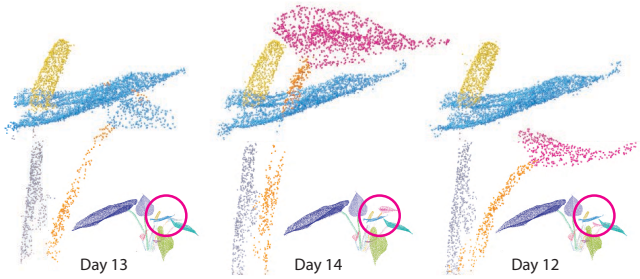

Figure 11: *A growing new leaf of the Anthurium plant. The new leaf starts with a very curvy shape, and touches the middle blue leaf on day 13. On day 14, it stretches and get detected as a new component, and further as a new leaf. Through backward analysis, this new leaf component helps segment the new leaf in day 13 and even go further towards its budding moment on day 12.*

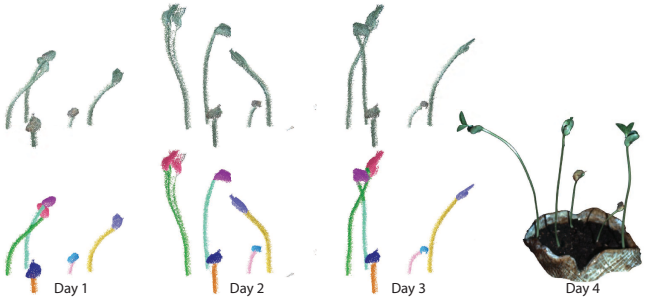

Figure 12: *Three frames of the dancing bean plant. The plant grows fast and exhibits large swaying motion during its development. Day 2 shows the blurring effect resulted from the dynamics within a single frame of the scanning period (minutes). These 'blurry' points are still correctly classified even though they exhibit quite different geometric features. On Day 3 that individual plants touch and intertwine, due to the large amount of motion. Our algorithm successfully handling all these scenarios using spatial and temporal coherence.*

As our method relies on temporal coherence, it could be sensitive if the plant demonstrates fast and large scale motion. Figures 12 shows three frames of the Dancing Bean plant where the stems swing by large amount (see accompanying video). The large scale motion makes geometric correspondence difficult to maintain. Our approach of first building geometric features, such as flatness and orientation, and then tracking structure correspondence based on feature analysis, works around this challenge. Note also the 'blur' of point cloud due to movements within the scanning period of one single frame. Branch extraction helps in data consolidation and together with the feature analysis, ensures building component correspondence between adjacent frames.

Plants at different development stages may have very different shapes. While the flatness feature works well with plants that have big leaves and thin stems, it does not work well with the soybeans that just break through the soil. In this case, we simply change the flatness feature into another thickness feature (radius of local points) in our dynamically thresholded forward-backward analysis.

Limitations. We investigated plants growing indoors in controlled glass-walled cubic space to ensure minimal external environmental disturbance (e.g., wind) and human interventions (e.g., trimming). Plants growing in open spaces can get more bushy and leafy, making the tracking of individual leaves almost impossible. Instead, the interest there would be tracking of major events, such as large branching and mass statistics.

We did not make use of any species-specific priors, which could improve both the accuracy (e.g., detecting a semi-folded leaf with the knowledge of the unfolding process inherent to the leaf development of certain species) and efficiency (e.g., detecting evidence of a new event early on) for the detection of an event. This would also help adapt parameters for our algorithm; for example, the flatness and orientation features, as used in the algorithm, can vary statistically from species to species, also at different stages development of a species.

While we leverage temporal coherence, it has a fairly 'short memory' due to its greedy nature, i.e., if part correspondence is lost for too long, it will not be re-tracked. This can happen, for example, when a leaf grows into surrounding leaves and branches, and grows out only after a long while. One solution can be to conduct backward search in large steps and employ global optimization when matching the structures of two time frames.

Our forward-backward analysis with adaptive reference values accommodates the characteristics of gradually evolving point features of growing plants, and is demonstrated to be effective for robust segmentation of plant organs. We observed that there are cases that two organs cannot be correctly segmented by features only. For example in Figure 11, the new leaf buds even before the moment we detect it with an extremely curvy shape and thus has features similar to stems. The prior that growing organs *gradually evolve in volume* can be incorporated to address this issue, i.e., by adding the constraint that the volume of each organ should change smoothly.

7 Applications

We demonstrate two applications with the 4D data and the information extracted from them: plant animation and simulation, both of which are very hard and tedious to be modelled by hand.

Synthesizing live plants with captured growth sequence. Our algorithm consistently segments 4D point cloud into individual organs, where the growth and motion of each organ can be extracted, as well as the relative bifurcation positions and times. Such information can be used to synthesize a live plant of similar species. Given a static plant model with proper correspondences to the organs extracted from the 4D point cloud data as the initial state, it can be brought to life by, or synthesized from, a real growth sequence (see Figure 1 and accompanying video). This is analogous to motion transfer widely used for human body, however, here not only the motion is captured/transferred, but also the growth.

Figure 13: *Robust leaf segmentation improves simulation. Individual frame based segmentation of Figure 10 (right) considers the two touching leaves as one resulting in two leaves glued together in the simulation (left). Cross frame organ information transfer identifies these two leaves and provides correct information for realistic simulation (right). The image is courtesy of [Zhao and Barbič 2013].*

Organ segmentation for simulation. Proper segmentation and hierarchical organization of the plant organs are critical for simulation (cf. [Zhao and Barbič 2013]). Since most plant models are built for rendering purpose only, additional effort has to be applied to author such models to be simulation-ready. Our algorithm outputs consistent segmentation of the 4D point cloud, resolves even organ touching or overlapping cases. The segmentation information in the point cloud is transfered to the reconstructed mesh model, making it segmented and hierarchically organized, and thus simulation ready (see Figure 13 and accompanying video). On the other hand, valuable properties can be extracted from the 4D data, and fed into plant simulators. For example, in Figure 14, the stiffness values of the stems are set as a function of their ages which are directly computed by our algorithm. The age based stiffness values help produce simulations that conform better to its observed motion in reality (see accompanying video).

Figure 14: *Extracted organ properties can be fed as simulation parameters. The semi-transparent blue stem is the initial position of the middle tall stem which is the youngest among all the stems and the most tender. A frame of the simulation with the stiffness of it set into 3 values: same as other stems (left), inversely proportional to its age (middle) and further exaggerated over the middle one (right). The age based stiffness setting produces simulations that conform better to its observed motions in reality.*

8 Conclusions

We presented a first attempt at capturing a developing plant and analyzing its evolving parts over time. This not only requires an elaborate setup for data acquisition, but also challenges existing state-of-the-art methods for 4D shape analysis. The challenges mainly lie in the departure from the incompressibility assumption in existing shape dynamics algorithms; here, new growth of shape components, or growth of new structures, constantly happens, in addition to the complexity caused by the changing point cloud density and occlusions among structures. The core of our approach is built on two premises: (i) temporal coherence inherent to plant growth, and (ii) increased discernibility of individual plant structures over a period of time. Our bi-directional approach fully takes advantage of the premises by carrying a new structure, detected robustly at a future moment, backward in locating its birth moment, and forward in propagating to future growth. We demonstrated our method by analyzing 4D point clouds of growing plants, recorded over days and in some cases weeks.

In the future, we plan to similarly analyze plant growth, but in outdoor settings. Beyond the challenges of data acquisition, we expect significant problems arising due to occlusion, limited accessibility, large scale plant movements (e.g., due to wind), outlier objects (e.g., birds, surrounding trees) in addition to detecting plant growth. Another venue of future work is incorporating color information into the analysis. Although color images are inherently 2D and are sensitive to change of lighting condition, they can nevertheless provide additional hints when point cloud data is severely degraded. Along this direction, we will also explore coupling additional physical sensors.

Acknowledgements

We are grateful to Zhan Song, Xufang Pang and Xiaoting Zhang for helping build the structured light scanning system, Jernej Barbič and Yili Zhao for running their plant simulator on our data, Sujuan Tan for taking care of the plants, Amit Shesh and Nathan Gossett for proofreading the paper. We would like to thank all the reviewers for their valuable comments.

This work is supported in part by grants from NSFC (61202221, 61232011, 61161160567), National 863 Program (2013AA01A604), Guangdong Science and Technology Program (2011B050200007), Shenzhen Innovation Program (CXB201104220029A, ZD201111080115A, KC2012JSJS0019A, KQCX20120807104901791, JCYJ20120617114842361), Israeli science foundation, Marie Curie CIG, and ERC Starting Grant SmartGeometry.

References

AHMED, N., THEOBALT, C., DOBREV, P., SEIDEL, H.-P., AND THRUN, S. 2008. Robust fusion of dynamic shape and normal capture for high-quality reconstruction of time-varying geometry. In *IEEE CVPR*, 1–8.

AKHTER, I., SIMON, T., KHAN, S., MATTHEWS, I., AND SHEIKH, Y. 2012. Bilinear spatiotemporal basis models. *ACM TOG 31*, 2, 17:1–17:12.

ALEXA, M., BEHR, J., COHEN-OR, D., FLEISHMAN, S., LEVIN, D., AND SILVA, C. T. 2001. Point set surfaces. In *IEEE Vis*, VIS '01, 21–28.

BEELER, T., HAHN, F., BRADLEY, D., BICKEL, B., BEARDSLEY, P., GOTSMAN, C., SUMNER, R. W., AND GROSS, M. 2011. High-quality passive facial performance capture using anchor frames. *ACM TOG 30*, 75:1–75:10.

BOJSEN-HANSEN, M., LI, H., AND WOJTAN, C. 2012. Tracking surfaces with evolving topology. *ACM TOG 31*, 4, 53:1–53:10.

BOYKOV, Y., AND KOLMOGOROV, V. 2004. An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision. *IEEE TPAMI 26*, 9, 1124–1137.

BOYKOV, Y., VEKSLER, O., AND ZABIH, R. 2001. Fast approximate energy minimization via graph cuts. *IEEE TPAMI 23*, 11, 1222–1239.

BRADLEY, D., POPA, T., SHEFFER, A., HEIDRICH, W., AND BOUBEKEUR, T. 2008. Markerless garment capture. *ACM TOG 27*, 3, 99:1–99:9.

BRENDEL, W., AND TODOROVIC, S. 2011. Learning spatiotemporal graphs of human activities. In *IEEE ICCV*, 778–785.

CHANG, W., AND ZWICKER, M. 2009. Range scan registration using reduced deformable models. *CGF 28*, 2, 447–456.

CHANG, W., AND ZWICKER, M. 2011. Global registration of dynamic range scans for articulated model reconstruction. *ACM TOG 30*, 3, 26:1–26:15.

CHEN, X., AND LAUX, T. 2012. Plant development - a snapshot in 2012. *Current Opinion in Plant Biology 15*, 1, 1 – 3.

CURLESS, B. 1999. From range scans to 3d models. *Proc. of SIGGRAPH 33*, 4 (Nov.), 38–41.

DE AGUIAR, E., STOLL, C., THEOBALT, C., AHMED, N., SEIDEL, H.-P., AND THRUN, S. 2008. Performance capture from sparse multi-view video. *ACM TOG 27*, 3, 98:1–98:10.

FERNANDEZ, R., DAS, P., MIRABET, V., MOSCARDI, E., TRAAS, J., VERDEIL, J.-L., MALANDAIN, G., AND GODIN, C. 2010. Imaging plant growth in 4d: robust tissue reconstruction and lineaging at cell resolution. *Nature Methods 7*, 7, 547–553.

GAUR, U., ZHU, Y., SONG, B., AND ROY-CHOWDHURY, A. 2011. A "string of feature graphs" model for recognition of complex activities in natural videos. In *IEEE ICCV*, 2595–2602.

HUANG, H., LI, D., ZHANG, H., ASCHER, U., AND COHEN-OR, D. 2009. Consolidation of unorganized point clouds for surface reconstruction. *ACM TOG 28*, 5, 176:1–176:7.

HUANG, H., WU, S., COHEN-OR, D., GONG, M., ZHANG, H., LI, G., AND CHEN, B. 2013. L1-medial skeleton of point cloud. *ACM TOG 32*.

KALAL, Z., MIKOLAJCZYK, K., AND MATAS, J. 2010. Forward-backward error: Automatic detection of tracking failures. In *Int. Conf. on Pattern Recognition*, 2756–2759.

KAZHDAN, M., BOLITHO, M., AND HOPPE, H. 2006. Poisson surface reconstruction. In *Proc. SGP*, 61–70.

KEVIN, T., FEI-FEI, L., AND KOLLER, D. 2012. Learning latent temporal structure for complex event detection. In *IEEE CVPR*.

KOLMOGOROV, V., AND ZABIH, R. 2004. What energy functions can be minimized via graph cuts? *IEEE TPAMI 26*, 2, 147–159.

LI, C., DEUSSEN, O., SONG, Y.-Z., WILLIS, P., AND HALL, P. 2011. Modeling and generating moving trees from video. *ACM TOG 30*, 6, 127:1–127:12.

LI, H., LUO, L., VLASIC, D., PEERS, P., POPOVIĆ, J., PAULY, M., AND RUSINKIEWICZ, S. 2012. Temporally coherent completion of dynamic shapes. *ACM TOG 31*, 1, 2:1–2:11.

LIAO, M., ZHANG, Q., WANG, H., YANG, R., AND GONG, M. 2009. Modeling deformable objects from a single depth camera. In *IEEE ICCV*, 167–174.

LIVNY, Y., YAN, F., OLSON, M., CHEN, B., ZHANG, H., AND EL-SANA, J. 2010. Automatic reconstruction of tree skeletal structures from point clouds. *ACM TOG 29*, 6, 151:1–151:8.

LU, C., CHELIKANI, S., JAFFRAY, D., MILOSEVIC, M., STAIB, L., AND DUNCAN, J. 2012. Simultaneous nonrigid registration, segmentation, and tumor detection in MRI guided cervical cancer radiation therapy. *IEEE Trans. on Medical Imaging 31*, 6, 1213–1227.

MITRA, N. J., FLÖRY, S., OVSJANIKOV, M., GELFAND, N., GUIBAS, L., AND POTTMANN, H. 2007. Dynamic geometry registration. In *Proc. SGP*, 173–182.

MÜNDERMANN, L., ERASMUS, Y., LANE, B., COEN, E., AND PRUSINKIEWICZ, P. 2005. Quantitative modeling of arabidopsis development. *Plant physiology 139*, 2, 960–968.

NEUBERT, B., FRANKEN, T., AND DEUSSEN, O. 2007. Approximate image-based tree-modeling using particle flows. *ACM TOG 26*, 3.

PIRK, S., NIESE, T., DEUSSEN, O., AND NEUBERT, B. 2012. Capturing and animating the morphogenesis of polygonal tree models. *ACM TOG 31*, 6 (Nov.), 169:1–169:10.

PIRK, S., STAVA, O., KRATT, J., SAID, M. A. M., NEUBERT, B., MĚCH, R., BENES, B., AND DEUSSEN, O. 2012. Plastic trees: interactive self-adapting botanical tree models. *ACM TOG 31*, 4 (July), 50:1–50:10.

PIRSIAVASH, H., AND RAMANAN, D. 2012. Detecting activities of daily living in first-person camera views. In *IEEE CVPR*.

PONS-MOLL, G., BAAK, A., GALL, J., LEAL-TAIXE, L., MULLER, M., SEIDEL, H.-P., AND ROSENHAHN, B. 2011. Outdoor human motion capture using inverse kinematics and von mises-fisher sampling. In *IEEE ICCV*, 1243–1250.

POPA, T., SOUTH-DICKINSON, I., BRADLEY, D., SHEFFER, A., AND HEIDRICH, W. 2010. Globally consistent space-time reconstruction. *CGF 29*, 5, 1633–1642.

PRUSINKIEWICZ, P., AND LINDENMAYER, A. 1996. *The algorithmic beauty of plants*.

PRUSINKIEWICZ, P., AND RUNIONS, A. 2012. Computational models of plant development and form. *New Phytologist 193*, 3, 549–569.

QUAN, L., TAN, P., ZENG, G., YUAN, L., WANG, J., AND KANG, S. B. 2006. Image-based plant modeling. *ACM TOG 25*, 3, 599–604.

ROZENBERG, G., AND SALOMAA, A. 1980. *Mathematical Theory of L Systems*. Academic Press, Inc., Orlando, FL, USA.

SHARF, A., ALCANTARA, D. A., LEWINER, T., GREIF, C., SHEFFER, A., AMENTA, N., AND COHEN-OR, D. 2008. Space-time surface reconstruction using incompressible flow. *ACM TOG 27*, 5, 110:1–110:10.

SHOTTON, J., FITZGIBBON, A., COOK, M., SHARP, T., FINOCCHIO, M., MOORE, R., KIPMAN, A., AND BLAKE, A. 2011. Real-time human pose recognition in parts from single depth images. In *IEEE CVPR*, 1297–1304.

SONG, Z., AND CHUNG, R. 2008. Use of lcd panel for calibrating structured-light-based range sensing system. *IEEE Trans. on Instrumentation and Measurement 57*, 11, 2623–2630.

SONG, Z., CHUNG, R., AND ZHANG, X.-T. 2013. An accurate and robust strip-edge-based structured light means for shiny surface micromeasurement in 3D. *IEEE Trans. on Industrial Electronics 60*, 3, 1023–1032.

TEVS, A., BERNER, A., WAND, M., IHRKE, I., BOKELOH, M., KERBER, J., AND SEIDEL, H.-P. 2012. Animation cartographyintrinsic reconstruction of shape and motion. *ACM TOG 31*, 2, 12:1–12:15.

THRUN, S., AND MONTEMERLO, M. 2005. The graphslam algorithm with applications to large-scale mapping of urban structures. *Int. J. on Robotics Research 25*, 5/6, 403–430.

VLASIC, D., BARAN, I., MATUSIK, W., AND POPOVIĆ, J. 2008. Articulated mesh animation from multi-view silhouettes. *ACM TOG 27*, 3, 97:1–97:9.

WAND, M., ADAMS, B., OVSJANIKOV, M., BERNER, A., BOKELOH, M., JENKE, P., GUIBAS, L., SEIDEL, H.-P., AND SCHILLING, A. 2009. Efficient reconstruction of nonrigid shape and motion from real-time 3d scanner data. *ACM TOG 28*, 2, 15:1–15:15.

XU, H., GOSSETT, N., AND CHEN, B. 2007. Knowledge and heuristic-based modeling of laser-scanned trees. *ACM TOG 26*, 4.

YAMAZAKI, S., NARASIMHAN, S. G., BAKER, S., AND KANADE, T. 2007. Coplanar shadowgrams for acquiring visual hulls of intricate objects. In *IEEE ICCV*, 1–8.

YAMAZAKI, S., NARASIMHAN, S. G., BAKER, S., AND KANADE, T. 2009. The theory and practice of coplanar shadowgram imaging for acquiring visual hulls of intricate objects. *IJCV 81*, 3, 259–280.

ZHAO, Y., AND BARBIČ, J. 2013. Interactive authoring of simulation-ready plants. *ACM TOG 32*, 4.

ZHENG, Q., SHARF, A., TAGLIASACCHI, A., CHEN, B., ZHANG, H., SHEFFER, A., AND COHEN-OR, D. 2010. Consensus skeleton for non-rigid space-time registration. *CGF 29*, 635–644.