

# Spatio-Temporal Non-Rigid Registration of 3D Point Clouds of Plants

Nived Chebrolu

Thomas Läbe

Cyrril Stachniss

**Abstract**—Analyzing sensor data of plants and monitoring plant performance is a central element in different agricultural robotics applications. In plant science, phenotyping refers to analyzing plant traits for monitoring growth, for describing plant properties, or characterizing the plant’s overall performance. It plays a critical role in the agricultural tasks and in plant breeding. Recently, there is a rising interest in using 3D data obtained from laser scanners and 3D cameras to develop automated non-intrusive techniques for estimating plant traits. In this paper, we address the problem of registering 3D point clouds of the plants over time, which is a backbone of applications interested in tracking spatio-temporal traits of individual plants. Registering plants over time is challenging due to its changing topology, anisotropic growth, and non-rigid motion in between scans. We propose a novel approach that exploits the skeletal structure of the plant and determines correspondences over time and drives the registration process. Our approach explicitly accounts for the non-rigidity and the growth of the plant over time in the registration. We tested our approach on a challenging dataset acquired over the course of two weeks and successfully registered the 3D plant point clouds recorded with a laser scanner forming a basis for developing systems for automated temporal plant-trait analysis.

## I. INTRODUCTION

Plant phenotyping [36] plays a fundamental role in the agricultural domain [10]. It involves the characterization of the plant traits, which helps plant breeders and other scientists to describe and to evaluate the overall plant performance. Traditionally, many of these plant traits are measured through time-consuming manual processes and are often intrusive as well. Recent studies [26], [19] showcase the potential of using 3D laser data for computing geometric plant traits with high quality. Agricultural robots equipped with such laser scanners can acquire 3D plant data at a large scale and facilitate high resolution phenotyping. This is a step forward towards scaling up phenotyping from plants grown in a greenhouse to the level of plots and maybe even fields.

To facilitate such a scale-up in phenotyping using 3D sensing, one of the fundamental requirements is the ability to register scans taken at different times. This bears quite some resemblance to the SLAM problem [31], loop closing, and ICP-based scan matching [5], often used in the context of laser-based SLAM. Similar to graph-based SLAM in non-static environments, we need to make data associations in changing scenes, align point clouds, and perform optimizations and iterative refinement procedures.

Registering plant scans recorded at different points in time, however, is a comparably challenging task due to



Fig. 1: A time-series of 3D point clouds of the same plant captured during its growth. Our goal is to develop techniques for registering such 3D scans captured under challenging conditions of changing topology and anisotropic growth of the plant.

the anisotropic growth of different organs of the plant, the change in its topology and the non-rigid deformations caused due to external stimuli such as wind or sunlight. In addition, the measurement process using laser scanners usually results in incomplete scans of the plant due to the numerous self-occlusion amongst leaves. Given these challenges, we aim at developing techniques, which facilitate the automatic computation of phenotypic traits from 3D time-series point clouds of plants.

Typically, point cloud registration is performed using iterative closest point-based approaches. These approaches, however, are often unable to capture the deformations in the object and are prone to divergence due to new or missing plant organs. In this paper, we investigate the means to account for the growth and the non-rigid deformations that the plant undergoes.

The main contribution of this paper is an iterative non-rigid registration process, which leverages the skeleton structure of the plant to obtain temporal matches between the scans. It determines correspondences using a hidden Markov model and estimate parameters, which are able to capture the deformation and the growth of the plant skeleton. We then transfer the deformations estimated on the plant skeletons to the whole point cloud in order to register the temporally separated point clouds. Using these registration parameters, we are also able to interpolate over the registration parameters to obtain an estimated point cloud at a time instant in-between the actual acquisition times.

In sum, our approach is able to (i) register temporally separated plant point clouds by explicitly accounting for the growth, deformations and changes in the plant topology, (ii) find correspondences between the different organs of the

plant, which allows for tracking plant growth parameters over time, (iii) demonstrate reliable registration results on a long-term dataset (recorded daily over two weeks). We illustrate this using the data of a tomato plant captured using a 3D laser scanner mounted on a robotic arm.

## II. RELATED WORK

Over the last decade, agricultural robotics has been receiving an increasing interest by the robotics community and the agricultural one alike. One relevant aspect in this context relates to obtaining relevant features of plants, often referred to as phenotypic traits, in an automated manner [10]. Several approaches [7], [9], [8] aim at obtaining traits at a coarse resolution over the entire field using image data captured from UAVs as well as from ground robots. More recently, 3D laser data has been analyzed to obtain phenotypic traits with high fidelity [2], [19], [26]. Li *et al.* [21] and Paproki *et al.* [25] analyze time-series point cloud data to detect topological events such as branching, decay and track the growth of different organs. While these works emphasize obtaining phenotypic traits at an organ level, our focus in this paper is on developing basic techniques for matching as well as registering temporally separated scans of individual plants using the whole point cloud data. This technique forms the basis for tracking phenotypic traits over time.

Several approaches in the past have attempted to leverage the topological structure of the object to drive the registration process, primarily in the field of human motion tracking [11], [15], [27]. A large corpus of literature exists for extracting skeletons from 3D models, which are then used for different applications such as animation, surface reconstruction etc. Huang *et al.* [17] and Tagliasacchi *et al.* [34] propose approaches to extract curve skeletons from unorganized point clouds which can be used as an input to our approach. A detailed state-of-the-art review for extracting skeletons of 3D objects is given in [33]. In this paper, we build upon a skeletal curve for the plant point clouds as an input and focus on using this skeletal structure of the plant to obtain correspondences reliably given the challenge of changing topology and growth of the organs.

Non-rigid registration techniques [6], [30], [32] aim at capturing the deformation in the object in addition to its motion while registering the scans. Approaches such as [13], [18], [20], [23], [24] aim at reconstructing scenes in an online fashion either in the presence of dynamic objects or deformations. Such approaches typically operate on scans captured at a high frame rate (10-30 Hz) and thereby deal with rather small deformations in between consecutive scans. In our application, the plants are usually scanned at a comparably low frequency (once per day) and thereby showing larger growth and deformations in between consecutive scans. Zheng *et al.* [37] propose an approach to register 3D temporal point clouds of objects by exploiting skeleton structures. We build upon their formulation and integrate it within an improved data association approach and iterative non-rigid registration scheme to register temporally separated 3D plant point clouds.

---

### Algorithm 1 Skeleton-driven iterative non-rigid registration

---

```

1:  $\mathcal{P}_1, \mathcal{P}_2$  ▷ Input point clouds
2:  $\mathcal{S}_1 \leftarrow \text{GETSKELETON}(\mathcal{P}_1)$ 
3:  $\mathcal{S}_2 \leftarrow \text{GETSKELETON}(\mathcal{P}_2)$ 
4:  $\mathcal{C}_{12}^{t-1}, \mathcal{C}_{12}^t = \emptyset$  ▷ Initialization
5: while  $(\mathcal{C}_{12}^t \setminus \mathcal{C}_{12}^{t-1}) \cup (\mathcal{C}_{12}^{t-1} \setminus \mathcal{C}_{12}^t) = \emptyset$  do
6:    $\mathcal{C}_{12}^{t-1} = \mathcal{C}_{12}^t$ 
7:    $\mathcal{C}_{12}^t \leftarrow \text{FINDSKELETALCORRESPONDENCES}(\mathcal{S}_1, \mathcal{S}_2)$ 
8:    $\mathcal{T}_{12} \leftarrow \text{COMPSKELETALDEFORMATION}(\mathcal{S}_1, \mathcal{S}_2, \mathcal{C}_{12}^t)$ 
9:  $\hat{\mathcal{P}}_1 \leftarrow \text{APPLYDEFORMATION}(\mathcal{P}_1, \mathcal{T}_{12})$ 

```

---

The problem at hand as well as the technique used in our approach to address plant registration shows some relation to techniques from the SLAM community [3], [4], [31]. This starts with the iterative scan alignment procedures such as ICP [5], [13], although we build upon a skeleton representation and not the point cloud itself. Furthermore, we build up data associations over time between skeleton nodes using a hidden Markov model formulation. For the optimization, least squares approaches related to graph-based SLAM [12] are used including robust kernel functions. Different to typical SLAM systems, however, we allow for multiple affine transformations between plant parts and thus extend the rigid-body transformations often found in the SLAM literature.

## III. OUR APPROACH TO PLANT REGISTRATION

Our approach operates on a time-series of consistent 3D point clouds of plants. Our registration starts with a skeleton for each point cloud and these skeletons can be computed using method described in [17]. The skeletons extracted are often imperfect and this fact is taken into consideration during the registration procedure. Note that we operate directly on the unordered point clouds and do not require a mesh structure or other cues such the normals providing the inside-outside information of the surface.

Our approach registers a point cloud pair  $(\mathcal{P}_1, \mathcal{P}_2)$  in an iterative manner. Similar to the popular ICP approach [5], we alternate between correspondence estimation steps and registration steps given the correspondences. In contrast to the nearest neighbor, point-to-plane, or normal-shooting correspondences used in typical ICP procedures, we use the skeletal structure of the plant to establish correspondences  $\mathcal{C}_{12}$ . This is done via a hidden Markov model (HMM) formulation detailed in Sec. III-A. Also in deviation from a standard ICP procedure, which assumes a rigid transformation between  $\mathcal{P}_1$  and  $\mathcal{P}_2$ , we explicitly model the deformation through different 3D affine transformations defined for each node of the skeleton  $\mathcal{S}_1$ . We estimate this set of affine registration parameters  $\mathcal{T}_{12}$  using a non-linear least squares procedure. We exit the iterative scheme when there is no change in the estimated correspondence set  $\mathcal{C}_{12}^t$ . After computing the registration parameters between the nodes of the skeletons  $\mathcal{S}_1$  and  $\mathcal{S}_2$ , we apply these parameters to the entire point cloud, see Sec. III-C. This results in the

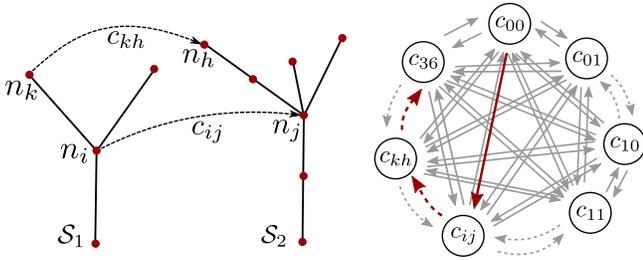


Fig. 2: Left: Example of skeletal matching with all the variables involved. Right: Hidden Markov model used for correspondence estimation. The red line depicts the sequence of best correspondence estimated by the Viterbi algorithm.

registered point clouds. The overall registration scheme is summarized in Alg. 1.

### A. Estimating Skeletal Correspondences

The goal of this step is to obtain correspondences between two skeletons of temporally separated scans. In order to cope with the imperfections in the individual skeletons and inconsistencies between them, we require a probabilistic approach to associate the skeleton parts as opposed to graph matching approaches, which do not tolerate such errors well. We therefore formulate the problem of finding correspondences between the skeleton pair  $(\mathcal{S}_1, \mathcal{S}_2)$  using a hidden Markov model formulation. The unknowns or the hidden states of the HMM model are all the potential correspondences between the nodes of the two skeletons. In addition, we also add a “not matched state” for each node to the HMM to account for the situations in which the node may have no correspondences at all. This happens, for example, when nodes belong to new organs that were not present before or when new nodes emerge on the curve skeleton due to the plant growth.

As required in a typical HMM formulation, we define the emission cost  $Z$  and the transition cost  $\Gamma$ . The emission cost  $Z$  describes the cost for a given hidden state (correspondence) to produce a certain observation. In our case, the observations are the sequence of nodes of the first skeleton  $\mathcal{S}_1$  arranged in depth first manner starting from the node at the base of the stem. We define this cost for a correspondence  $c_{ij} \in \mathcal{C}_{12}$  between node  $n_i$  of  $\mathcal{S}_1$  and node  $n_j$  of  $\mathcal{S}_2$  as:

$$Z(c_{ij}) = w_d |deg(n_i) - deg(n_j)| + w_e \|x_i - x_j\|, \quad (1)$$

where the first term yields the difference between the degrees of the corresponding nodes, where  $deg(n)$  is the number of edges incident to a node. The second term refers to the Euclidean distance between them with  $(x_i, x_j)$  being the 3D locations of the nodes  $(n_i, n_j)$  respectively. The idea behind combining these two terms is capture both, the topology difference as well as actual spatial distance between the nodes. This combined cost will be smaller for correspondences between nodes that have similar topology and are located close to each other. We weigh the two terms using  $w_d$  and  $w_e$  to properly scale the two measures.

The transition cost  $\Gamma$  describes the cost involved in transitioning from one hidden state  $c_{ij}$  to another  $c_{kh}$ . This can be

treated as the cost involved in having  $c_{kh}$  as a valid match given that  $c_{ij}$  is a valid match as well. We define this cost as:

$$\begin{aligned} \Gamma(c_{ij}, c_{kh}) = & |d_g(n_i, n_k) - d_g(n_j, n_h)| \\ & + w_{nbr} |n_{br}(n_i, n_k) - n_{br}(n_j, n_h)| \\ & + \rho_{dir}((x_i - x_j), (x_k - x_h)), \end{aligned} \quad (2)$$

where the first term computes the difference of the geodetic distances  $d_g$  between the nodes involved in the two correspondence pairs along their respective skeletons. This means that a pair of correspondences  $(c_{ij}, c_{kh})$  having equal geodetic lengths  $d_g(n_i, n_k)$  along  $\mathcal{S}_1$  and  $d_g(n_j, n_h)$  along  $\mathcal{S}_2$  will have a lower cost than the ones which have much different lengths along the skeleton. The second term captures the difference in the number of branches  $n_{br}$ , i.e., nodes with degree greater than 2, along the way on the skeleton. The weight  $w_{nbr}$  is automatically set as the maximal geodetic distance between all node pairs of the first skeleton. The final term  $\rho_{dir}$  is a function that penalizes the correspondence pairs  $(c_{ij}, c_{kh})$  with a large cost if the directions determined by  $(x_i - x_j)$  and  $(x_k - x_h)$  are opposite, i.e., the angle between them are greater than  $\frac{\pi}{2}$ .

Once the emission and transition costs are defined, we compute the correspondences between the skeletons by performing an inference on the HMM. The result is the most likely sequence of hidden variables, i.e., the set of correspondences between  $\mathcal{S}_1$  and  $\mathcal{S}_2$ . We perform this inference using the Viterbi algorithm [35]. In case a node has more than one correspondence, we choose the correspondence with the smaller Euclidean distance to ensure a one-to-one correspondence. As an illustration, Fig. 2 (left) shows an example skeleton pair for which we want to estimate the correspondences  $\mathcal{C}_{12}$ . Fig. 2 (right) depicts the HMM for the example pair where the red path indicates the set of correspondences estimated by the Viterbi algorithm. The HMM model only shows a sub-set of the connections between the hidden states, where in practice each state is connected to every other state.

### B. Computing Skeletal Deformation Parameters

In this step, we compute the registration parameters between  $\mathcal{S}_1$  and  $\mathcal{S}_2$  given the set of correspondences  $\mathcal{C}_{12}$ . While registering temporally separated plant scans, the shape and the topology of the plant changes. Therefore, to capture these changes we need to forego the usual assumption of rigidity often used in point cloud registration. Our goal is to capture the non-rigid changes by computing sets of deformation parameters between skeleton parts of the respective plant scans. We estimate these deformation parameters through a non-linear least squares optimization procedure based on the correspondences obtained from the procedure described in the previous section.

To model the deformations between the plant scans, we attach an affine transformation  $T_i$  to each node  $n_i$  of the skeleton  $\mathcal{S}_1$ . The intuition behind such a model is that the skeleton may be deformed differently at different locations

along the skeleton. By modeling the deformations through a 3D affine transformation with 12 unknown parameters per node, we are able to capture the growth as well as bending of the plant via the scaling, shearing, and rotation parameters.

We define the objective function of the optimization problem as a combination of the three energy terms. The first term  $E_{corresp}$  is defined as:

$$E_{corresp} = \sum_{c_{ij} \in \mathcal{C}_{12}} \|T_i x_i - y_j\|, \quad (3)$$

where  $x_i$  and  $y_j$  are the node positions given by the correspondence pair  $c_{ij}$  estimated in Sec. III-A. This energy term captures the distance between corresponding nodes in  $\mathcal{S}_1$  and  $\mathcal{S}_2$  and strives to make this error as small as possible during optimization.

The second energy term  $E_{rot}$  captures how close the estimated affine transformation is to a pure rotation and it determines the smoothness of the deformation. We define  $E_{rot}$  as:

$$E_{rot} = \sum_{\substack{i=1 \\ j=\text{mod}(i+1,3)}}^3 (c_i c_j)^2 + \sum_{i=1}^3 (c_i c_i - 1)^2, \quad (4)$$

where  $c_i$  represents the columns of the rotation part of affine transformation (i.e. the first three rows and columns of  $T_i$ ). The first term in  $E_{rot}$  in Eq. (4) measures the deviation for a pair of columns to be orthogonal with each other, whereas the second term measures the deviation of each column from being unit length.  $E_{rot}$  forces the estimated affine parameters  $T_i$  to be as close to a true rotation as possible.

We also define a regularization term  $E_{reg}$  as:

$$E_{reg} = \sum_{j \in N(i)} \|T_i^{-1} T_j - I\|, \quad (5)$$

where  $T_i, T_j$  are transformations corresponding to nodes  $n_i, n_j$  such that  $j$  is the neighbor  $N(i)$  along  $\mathcal{S}_1$ .  $E_{reg}$  is a regularizing term, which forces the transformation parameters of neighboring nodes to be similar. This results in a smooth deformation along the skeleton and achieves similar results as the as-rigid-as-possible constraint described by Sorkine *et al.* [30]. The regularization term is also necessary to constrain the nodes that do not have any correspondences. Finally, the combined energy  $E_{total}$  is obtained as a weighted combination of all the three energies as:

$$E_{total} = w_{corresp} E_{corresp} + w_{rot} E_{rot} + w_{reg} E_{reg} \quad (6)$$

We use the weights  $w_{corresp} = 100$ ,  $w_{rot} = 10$ , and  $w_{reg} = 1$  for all the example in our datasets. The weights have been chosen such that the cost due to each component of the loss is in the same order of magnitude. We employ the standard Gauss-Newton algorithm to solve the unconstrained non-linear least squares problem [14]. We use the Cauchy robust kernel [22] for the error residuals belonging to  $E_{corresp}$  as this prevents incorrect correspondences to influence the optimization process. The approach is related to the formulation by Sumner *et al.* [32] for

estimating deformation parameters for surfaces parametrized as triangular meshes. In our case, we adapt the energy terms to reflect the constraints valid for deformation between curve skeletons as opposed to surfaces. Furthermore, the approach in [32] is not able to fully constrain the nodes which have a degree smaller than 3, but is essential for registration of curve skeletons.

### C. Point Cloud Deformation

To obtain the final registered point cloud  $\mathcal{P}'_1$ , we need to apply the deformation parameters estimated for the skeleton nodes in Sec. III-B to all the points of the scan. For each point  $p$  in the point cloud  $\mathcal{P}_1$ , we obtain the deformed point  $p'$  as a weighted sum of affine transformations corresponding to the two nearest nodes to the point  $p$  as

$$p' = \sum_{k \in N(p)} \alpha_k T_k p, \quad (7)$$

where  $k$  is the index of the nearest node  $N(p)$  and  $\alpha_k$  is computed according to the projection of the point  $p$  on the edge of the skeleton determined by the nearest nodes. If  $p_e$  is the projection of point  $p$  on edge  $e$ . Then,

$$\alpha_k = 1 - \|p - e\|/\|e\|. \quad (8)$$

### D. Interpolating Point Clouds

In addition to registering the plant scans recorded at different times, we would also like to interpolate how the plant may be deformed at an intermediate time in between the actual acquisition times. We compute the deformed point cloud by interpolating the deformation parameters  $T$  estimated between the two registered scans. To obtain a smooth interpolation, we first decompose of the estimated affine transformation  $T$  into scale/shear transformation  $T_s$ , pure rotation  $T_R$ , and a pure translation  $T_t$  using the polar decomposition approach described in [29].

$$T = T_s T_R T_t \quad (9)$$

We then linearly interpolate  $T_s$  and  $T_t$  to obtain the transformation at time  $t$ . For interpolating  $T_R$ , we use the spherical linear interpolation described in [28].

## IV. EXPERIMENTAL EVALUATION

### A. Dataset Description

We evaluate our approach on a time-series 3D point cloud data of a tomato plant (*Solanum lycopersicum*). The scans were recorded using a robotic arm (Romer Absolute Arm) [1] equipped with a high precision laser scanner. The dataset was recorded daily over two weeks. The plants have been scanned to minimize self occlusions whenever possible. Also all points not belonging to the plant have been removed in a pre-processing step. The dataset covers the growth of the plant from the 2 leaves stage to around 10 leaves capturing several branching events and substantial leaf growth till the end of the acquisition period. We obtain the skeletons from the 3D point clouds using the algorithm described in [17].

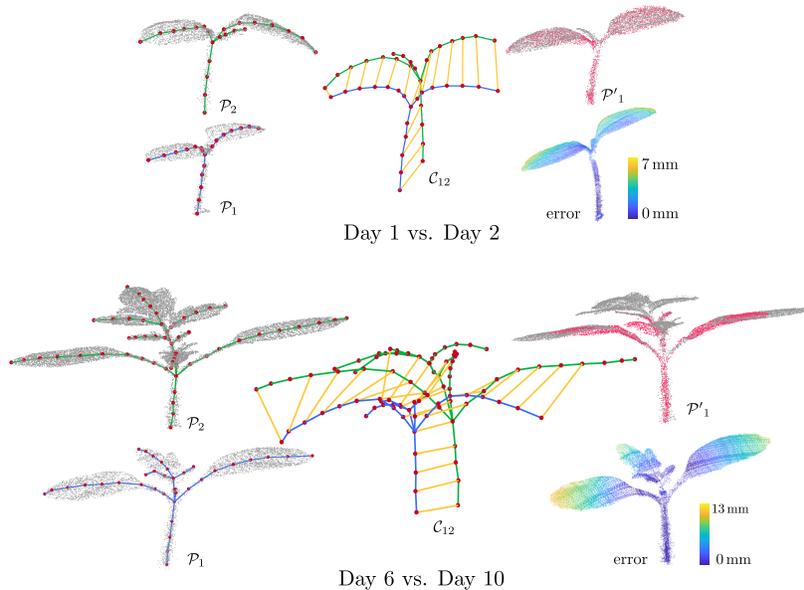


Fig. 3: 4D registration of point clouds. Top: shows registration results between scans from consecutive days (Day 1 and Day 2). Bottom: shows registration results between scans which are 4 days apart (Day 6 and Day 10). The left column shows the two input point clouds ( $\mathcal{P}_1, \mathcal{P}_2$ ) along with their skeletons, middle column shows the estimated correspondences (yellow lines) between the skeletons, and the right column shows the deformed point cloud  $\mathcal{P}'_1$  (in pink) overlaid on  $\mathcal{P}_2$  along with registration error visualized as a heat map.

### B. 4D Registration of Point Clouds

The first experiment is designed to illustrate the results of our plant registration pipeline for time-series point cloud data of the plants and to quantitatively evaluate the accuracy of the registration pipeline. Fig. 3 illustrates the results of the registration procedure for two example scan pairs. The first example (Fig. 3 Top) visualizes the registration results for scans from consecutive days, whereas the second example (Fig. 3 Bottom) shows registration between scans which are farther apart (4 days here). For both examples, we show the input point clouds ( $\mathcal{P}_1, \mathcal{P}_2$ ) along with their corresponding skeletons. The correspondences estimated during the registration procedure are depicted by the yellow-lines joining the nodes of the skeleton pair. Our approach was able to find the correspondences reliably despite the growth and the change in topology, which is specially prominent in the second example (Fig. 3 Bottom). We visualize the final registered point cloud  $\mathcal{P}'_1$  (in pink) by deforming the point cloud  $\mathcal{P}_1$  using the deformation parameters estimated by our approach and overlay it on the target point cloud  $\mathcal{P}_2$  (in gray) and observe that it overlaps well indicating that the registration results are reasonable.

Further, we quantitatively evaluate the accuracy of our registration pipeline by registering all consecutive scans of the dataset. First, we compute the accuracy of our skeleton matching procedure by computing the percentage of correspondences estimated correctly. We define the correct correspondences as those which belong to the same organ (i.e., the same leaf or the stem) in the skeleton pair as there is no unique way to define a correct correspondence due to the growth in the plant. We manually label the different organs of the plant with a unique identifier to provide the ground

truth to compute this metric. For our dataset, we obtain an average of 95% correct correspondences between consecutive skeleton pairs with most pairs having all the correspondences estimated correctly.

Secondly, we evaluate the accuracy of the estimated registration parameters by computing the error between the deformed source point cloud  $\mathcal{P}'_1$  and the target point cloud  $\mathcal{P}_2$ . We define this registration error  $e_{reg}$  as:

$$e_{reg} = \frac{1}{|\mathcal{P}'_1|} \sum_{\substack{i=1 \\ j \in \mathcal{N}(i)}}^{|\mathcal{P}'_1|} \|p'_1{}^i - p_2^j\|, \quad (10)$$

where  $p_2^j$  is the nearest point to  $p'_1{}^i$  and  $|\mathcal{P}'_1|$  is the number of points in  $\mathcal{P}'_1$ .

For our dataset, we obtain a mean error of 3 mm and a maximum error of 13 mm for consecutive scans, which indicates that the registration results are accurate. As a baseline comparison, we computed the average overlap error by assuming a rigid transformation between the scans and obtain an average error  $e_{reg}$  of 35 mm and maximum error of 166 mm. The large errors using a rigid transformation assumption are both due to the plant growth and in some cases the ICP procedure diverging completely. This also indicates that a rigid transformation assumption is inadequate and a non-rigid registration procedure is required to capture the growth and movement of the plant.

We visualize the registration error as a heat map for the two example point cloud pairs in Fig. 3 (Bottom right of each example). The heat map is projected on  $\hat{\mathcal{P}}_1$  to show how well different portions of the plant are registered. The blue regions in the heat map represent a smaller registration error whereas the yellow regions indicate large errors. Most

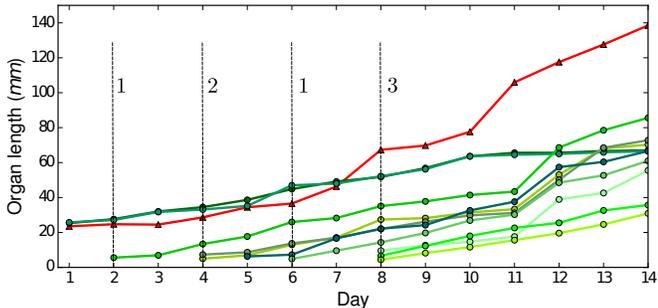


Fig. 4: Estimating growth for individual organs. Our registration procedures allows us to track the growth of the stem and different leaf lengths over time and detect topological events such as the emergence of new leaves.

of the regions are blue indicating a successful registration, however, we notice that the errors are usually high towards the outer sections of the leaves which are farther away from the skeleton curve. This effect is to be expected as the skeleton curves do not capture this area well.

### C. Growth Estimation

The second experiment is designed to illustrate that we can derive the growth information about the plant given the spatial-temporal registration of our approach. We leverage the registration information between the plant scans and automatically extract some growth parameters as well as other important events, which helps us to monitor certain aspects of the plant growth. As a typical phenotypic trait example, we compute individual leaf, and stem lengths and monitor their growth over the acquisition period. This growth is visualized in Fig. 5, where the red line represents the stem length and the different shades of green lines represent the length of the corresponding leaves over time. In addition, we also detect certain events which mark a topological change in the structure of the plant such as the appearance of a new leaf. We show the number of new leaves emerging each day next to the dashed vertical lines (e.g. three new leaves emerge on day 8). In Fig. 5, we see how a leaf first appears on day 2 and then grows up to 40 mm by day 11, and then in another phase of rapid growth reaches to about 80 mm by day 14. Such information can be used to compute the BBCH scale [16] of the plant which is a growth stage scale and other time varying phenotypic traits of a plant desired by agronomists. The computation of such traits is only possible due to the fact that the scans are temporally registered against each other.

### D. Temporal Interpolation of Point Clouds

In this last experiment, we showcase that we can even interpolate the point clouds at intermediate points in time, i.e., in between the actual acquisition of the scans. The ability to interpolate is advantageous for analyzing properties of the plant scans even when actual measurements are not available. It allows us to predict both, the motion and growth at intermediate time intervals. We visualize the interpolated point cloud at  $t_i$  mid-way between the two scans in top

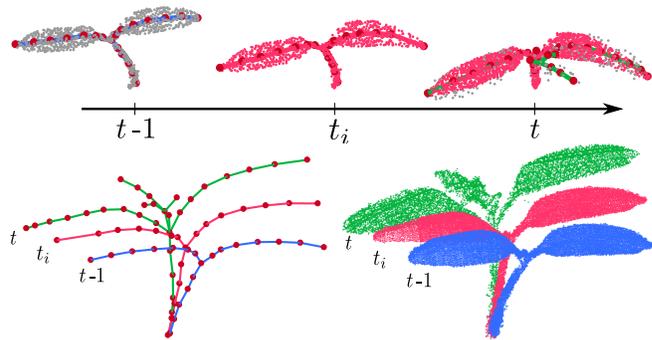


Fig. 5: Top: interpolation of point clouds at an intermediate time interval. Point clouds (gray) at time  $t - 1$  and  $t$  come from actual scan measurements whereas the points cloud (pink) at in-between instant  $t_i$  is the interpolated scan. Bottom left: shows the skeletons at time  $t - 1$  (blue),  $t_i$  (pink) and  $t$  (green). Bottom right: shows the corresponding point clouds in the same color as the skeletons. We see that the interpolated skeleton and the point cloud (in pink) captures the growth well between  $t - 1$  and  $t$ .

of Fig. 5. This allows us to animate a time-lapse view of the plants. The pink point clouds represent the interpolated scans and overlaps well with point cloud (gray) at time  $t$  indicating that the interpolation is reasonable. To visualize the growth captured by interpolation better, we show the skeletons and the point clouds at times  $t - 1$ ,  $t_i$ ,  $t$  together (Fig. 5 Bottom). As the interpolation procedure does not actually model the movement or the growth of the plant, the result of the interpolation may differ from the actual plant at those instances. In order to evaluate the interpolation step, we take the scans on day  $t - 1$  and day  $t + 1$ , then interpolate the point cloud at day  $t$  and compare against the actual point cloud on day  $t$ . We compute the registration error (as described in 10) and obtain a mean  $e_{reg}$  of 4 mm suggesting that our interpolation is a reasonable approximation of the real plant growth.

## V. CONCLUSION

In this paper, we presented a novel approach for spatio-temporal registration of 3D point clouds of individual plants. Our method builds upon the skeletal structure of the plant to find reliable correspondences between parts of the plant using an HMM-based approach. Furthermore, we explicitly model the deformation and growth of the plant over time using multiple affine transformations. We implemented and evaluated our approach on a long-term dataset of a tomato plant presenting challenging situations and supported all claims made in this paper. The experiments suggest that our registration approach can be used as a basis for tracking plant traits temporally and contribute towards automated phenotyping. As future work, we aim to evaluate the approach on different types of plants and extend it deal with more noisy scans obtained from mobile platforms.

## ACKNOWLEDGMENTS

We thank the Institute for Geodesy and Geoinformation (IGG), University of Bonn for providing the 3D scans of the tomato plant.

## REFERENCES

- [1] Romer absolute arm. <https://www.hexagonmi.com/products/portable-measuring-arms>.
- [2] G. Aleny, B. Dellen, and C. Torras. 3d modelling of leaves from color and tof data for robotized plant measuring. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, pages 3408 – 3414, 06 2011.
- [3] T. Bailey and H.F. Durrant-Whyte. Simultaneous localisation and mapping (SLAM): Part I. *IEEE Robotics and Automation Magazine (RAM)*, 13(2):99–110, 2006.
- [4] T. Bailey and H.F. Durrant-Whyte. Simultaneous localisation and mapping (SLAM): Part II. *IEEE Robotics and Automation Magazine (RAM)*, 13(3):108 –117, 2006.
- [5] P.J. Besl and N.D. McKay. A Method for Registration of 3D Shapes. *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, 14(2):239–256, 1992.
- [6] S. Bouaziz, A. Tagliasacchi, H. Li, and M. Pauly. Modern techniques and applications for real-time non-rigid registration. In *SIGGRAPH ASIA 2016 Courses*, SA '16, pages 11:1–11:25, New York, NY, USA, 2016. ACM.
- [7] L. Carlone, J. Dong, S. Fenu, G.G. Rains, and F. Dellaert. Towards 4d crop analysis in precision agriculture: Estimating plant height and crown radius over time via expectation-maximization. In *ICRA Workshop on Robotics in Agriculture*, 2015.
- [8] N. Chebrolu, T. Labe, and C. Stachniss. Robust long-term registration of uav images of crop fields for precision agriculture. *IEEE Robotics and Automation Letters*, 3(4):3097–3104, 2018.
- [9] J. Dong, J.G. Burnham, B. Boots, G. Rains, and F. Dellaert. 4D Crop Monitoring: Spatio-Temporal Reconstruction for Agriculture. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2017.
- [10] R. T. Furbank and M. Tester. Phenomics technologies to relieve the phenotyping bottleneck. *Trends in Plant Science*, 16(12):635 – 644, 2011.
- [11] J. Gall, C. Stoll, E. De Aguiar, C. Theobalt, B. Rosenhahn, and H.P. Seidel. Motion capture using joint skeleton tracking and surface estimation. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1746–1753, 2009.
- [12] G. Grisetti, R. Kummerle, C. Stachniss, and W. Burgard. A tutorial on graph-based SLAM. *IEEE Trans. on Intelligent Transportation Systems Magazine*, 2:31–43, 2010.
- [13] D. Haehnel, S. Thrun, and W. Burgard. An extension of the icp algorithm for modeling nonrigid objects with mobile robots. In *Proc. of the Intl. Conf. on Artificial Intelligence (IJCAI)*, volume 3, pages 915–920, 2003.
- [14] M. T. Heath. *Scientific Computing: An Introductory Survey*. McGraw-Hill Higher Education, 2nd edition, 1996.
- [15] L. Herda, P. Fua, R. Plankers, R. Boulic, and D. Thalmann. Skeleton-based motion capture for robust reconstruction of human motion. In *Proceedings Computer Animation 2000*, pages 77–83, May 2000.
- [16] M Hess, G Barralis, H Bleiholder, L Buhr, TH Eggers, H Hack, and R Stauss. Use of the extended bbch scalegeneral for the descriptions of the growth stages of mono; and dicotyledonous weed species. *Weed Research*, 37(6):433–441, 1997.
- [17] H. Huang, S. Wu, D. Cohen-Or, M. Gong, H. Zhang, G. Li, and B. Chen. L1-medial skeleton of point cloud. *ACM Trans. on Graphics (TOG)*, 32(4):65–1, 2013.
- [18] M. Innmann, M. Zollhofer, M. Niener, C. Theobalt, and M. Stamminger. VolumeDeform: Real-time Volumetric Non-rigid Reconstruction. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, pages 362–379, 2016.
- [19] R. Klose, J. Penlington, and A. Ruckelshausen. Usability study of 3d time-of-flight cameras for automatic plant phenotyping. *Bornimer Agrartechnische Berichte*, 69(93-105):12, 2009.
- [20] H. Li, R.W. Sumner, and Mark Pauly. Global correspondence optimization for non-rigid registration of depth scans. In *Computer graphics forum*, volume 27, pages 1421–1430. Wiley Online Library, 2008.
- [21] Y. Li, X. Fan, N.J. Mitra, D. Chamovitz, D. Cohen-Or, and B. Chen. Analyzing growing plants from 4d point cloud data. *ACM Transactions on Graphics*, 32(6):157, 2013.
- [22] K. MacTavish and T. D. Barfoot. At all costs: A comparison of robust cost functions for camera correspondence outliers. In *2015 12th Conference on Computer and Robot Vision*, pages 62–69. IEEE, 2015.
- [23] R.A. Newcombe, D. Fox, and S.M. Seitz. DynamicFusion: Reconstruction and Tracking of Non-Rigid Scenes in Real-Time. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 343–352, 2015.
- [24] E. Palazzolo, J. Behley, P. Lottes, P. Giguere, and C. Stachniss. ReFusion: 3D Reconstruction in Dynamic Environments for RGB-D Cameras Exploiting Residuals. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2019.
- [25] A. Paproki, X. Sirault, S. Berry, R. Furbank, and J. Fripp. A novel mesh processing based technique for 3d plant analysis. *BMC plant biology*, 12(1):63, 2012.
- [26] S. Paulus, H. Schumann, H. Kuhlmann, and J. Leon. High-precision laser scanning system for capturing 3D plant architecture and analysing growth of cereal plants. *Biosystems Engineering*, 121:1–11, 2014.
- [27] L.A. Schwarz, A. Mkhitarian, D. Mateus, and N. Navab. Human skeleton tracking from depth data using geodesic distances and optical flow. *Image and Vision Computing*, 30(3):217 – 226, 2012.
- [28] K. Shoemake. Animating rotation with quaternion curves. *Proc. of the Intl. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pages 245–254, 1985.
- [29] K. Shoemake and T. Duff. Matrix animation and polar decomposition. In *Proc. of the Conf. on Graphics Interface*, volume 92, pages 258–264, 1992.
- [30] O. Sorkine and M. Alexa. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, volume 4, pages 109–116, 2007.
- [31] C. Stachniss, J. Leonard, and S. Thrun. *Springer Handbook of Robotics, 2nd edition*, chapter Chapt. 46: Simultaneous Localization and Mapping. Springer Verlag, 2016.
- [32] R. W. Sumner, J. Schmid, and M. Pauly. Embedded deformation for shape manipulation. *ACM Trans. on Graphics (TOG)*, 26(3):80, 2007.
- [33] A. Tagliasacchi, T. Delame, M. Spagnuolo, N. Amenta, and A. Telea. 3d skeletons: A state-of-the-art report. In *Computer Graphics Forum*, volume 35, pages 573–597. Wiley Online Library, 2016.
- [34] A. Tagliasacchi, H. Zhang, and D. Cohen-Or. Curve skeleton extraction from incomplete point cloud. In *ACM Trans. on Graphics (TOG)*, volume 28, page 71. ACM, 2009.
- [35] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Trans. on Information Theory*, 13(2):260–269, 1967.
- [36] A. Walter, F. Liebisch, and A. Hund. Plant phenotyping: from bean weighing to image analysis. *Plant Methods*, 11(1), 2015.
- [37] Q. Zheng, A. Sharf, A. Tagliasacchi, B. Chen, H. Zhang, A. Sheffer, and D. Cohen-Or. Consensus skeleton for non-rigid space-time registration. In *Computer Graphics Forum*, volume 29, pages 635–644. Wiley Online Library, 2010.