Towards In-Field Phenotyping Exploiting Differentiable Rendering with Self-Consistency Loss

Federico Magistri Nived Chebrolu Jens Behley Cyrill Stachniss

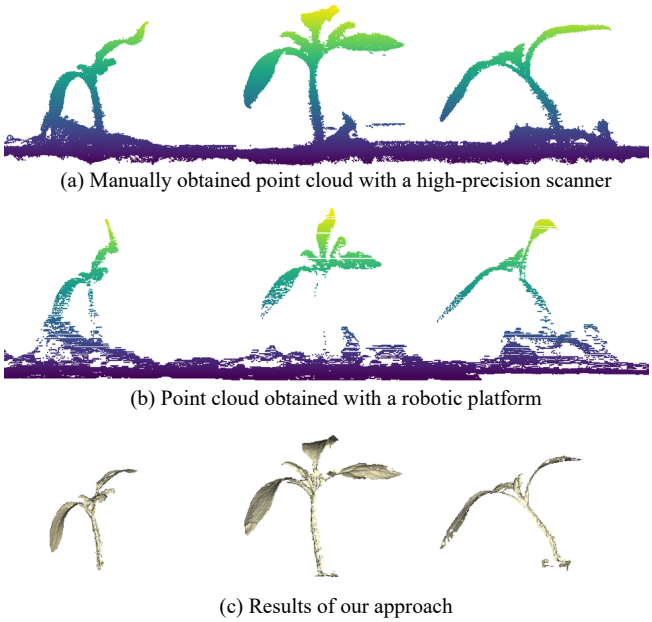
***Abstract*— In modern agriculture, measuring phenotypic traits helps breeders monitor plant growth, increase yield, and provide food, feed, and fiber. Traditional phenotyping requires intensive manual work, partially being intrusive. In this paper, we investigate the challenge of measuring phenotypic traits in an automated fashion through mobile robots operating in field environments. In particular, we want to measure plants from images acquired by mobile robots instead of using data from a static scanning environment. We propose to use a differentiable rendering approach to deform a generic 3D template of a plant to fit the observation recorded by a robot while ensuring a coherent deformation of the plant template. The experiments presented in this paper suggest that our approach allows for 3D reconstruction of different plant species at different growth stages using single images. From that model, we can compute important phenotypic traits, such as the leaf area index.**

I. Introduction

Phenotyping is the task of measuring plant traits to describe plant physiology and it is central in plant science. Also breeders use phenotyping measurements to support decisions in crop fields. Such decisions include selecting the best cultivars to continue the breeding process and selecting the best cultivars for the following seasons. Phenotyping, however, is expensive, time-consuming, and requires intrusive operations that potentially damages the crop. Phenotyping is mostly performed during the plant breeding process, and in this context, mobile robots equipped with sensors and data interpretation capabilities have the potential to become a game-changer [8], [7].

In recent years, there has been an increase in studies employing robots in fields. The majority of those works exploit sensor data to tackle tasks such as weeding [21], [10], [25], [36], crop counting [18], [39], or fruit picking [19]. Fewer works exploit robots to measure phenotypic traits based on the complete 3D plant geometry, which are important to evaluate crop health. Existing approaches often measure only basic traits such as crop height [3], [4] when operating in the field and outside greenhouses.

This paper tackles estimating important phenotypic traits, such as the leaf area, through mobile robots, using regular 2D camera images instead of costly 3D reconstruction settings. Measuring the leaf area, for example, is an important estimator of a plant's capability to capture sunlight. Current approaches employ a flatbed scanner to measure the leaf area. This approach is not suited for large-scale monitoring since humans have to measure each leaf in the field manually. In the past years, several studies presented a way to measure the leaf areas. These approaches, however, require highly accurate 3D point clouds of plants to measure such phenotypic traits [41], [40], [9], [11].



(a) Manually obtained point cloud with a high-precision scanner

(b) Point cloud obtained with a robotic platform

(c) Results of our approach

Fig. 1: To measure phenotypic traits, it is fundamental to obtain 3D data of plants with high spatial resolution. Yet, state-of-the-art approaches, either registration or reconstruction algorithms, are not sufficient for accurate phenotyping at a plant level. As an example, the top row shows the point cloud obtained with a hand-held laser scanner, the middle row shows the same scene captured with a depth camera. Our goal is to infer the 3D geometry of plants using single images shown in the bottom row depicting the triangle meshes reconstructed with our approach.

In the agricultural setting, obtaining high-fidelity point clouds is challenging due to the high level of details needed. For example, Fig. 1 illustrates the differences between a point cloud obtained with a high-precision laser scanner and with human support (top row) and a point cloud obtained with a robotic onboard depth camera (middle row). One can see the challenges of in-field phenotyping, such as noisy and incomplete data that make the phenotypic evaluation of traits at a plant level inaccurate. Our goal is to recover the 3D geometry of plants using only 2D images and a database of 3D templates of generic plants. The results of our approach are shown in the bottom row.

All authors are with the University of Bonn, Germany.

This work has been funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy, EXC-2070 – 390732324 (PhenoRob) and by the Federal Ministry of Food and Agriculture (BMEL) based on a decision of the Parliament of the Federal Republic of Germany via the Federal Office for Agriculture and Food (BLE) under the innovation support programme under funding no 28DK108B20 (RegisTer).

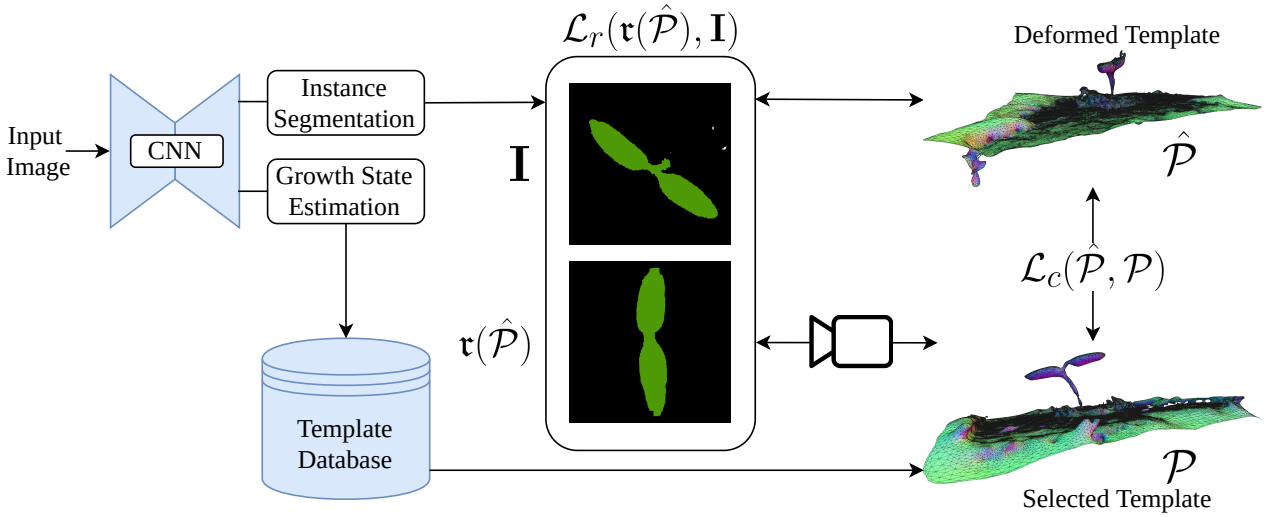


Fig. 2: An overview of our approach. We use as a preprocessing step the work of Weyler et al. [35] that provides a pixel-wise instance segmentation and an estimate of the BBCH index. Based on this estimate, we select a plant template $\mathcal{P}$, in the form of a triangle mesh, from our database. Our goal is to deform the template, $\hat{\mathcal{P}}$, such that it fits the current observation from the robot, I. This is possible by combining a differentiable rendering module with notions borrowed from non-rigid registration literature. We first render our template mesh, denoted as $\mathfrak{r}(\hat{\mathcal{P}})$, and then we deform it such that the rendered image $\mathfrak{r}(\hat{\mathcal{P}})$ and the input image I look similar, which is guided by the reconstruction loss $\mathcal{L}_r$. Additionally, we enforce local rigidity with a structural loss $\mathcal{L}_c$ that aims at maintaining the structure of the plant template $\mathcal{P}$ with the deformed mesh $\hat{\mathcal{P}}$.

The main contribution of this paper is a method to infer the 3D shape of plants using single 2D images. We exploit the prior knowledge about the structure of plants [28] at a given growth stage called BBCH index [12], by computing a library of generic plant templates that we then modify. Given such a library and the taken image of a plant, we select the most appropriate template based on the BBCH index. We, then, deform the selected plant template, with a differentiable rendering approach, such that its rendered view aligns with the target image. In this way, we can estimate phenotypic traits, such as the leaf area index (LAI) without the need for costly 3D reconstruction nor human intervention.

In sum, our approach can (i) compute a simplified mesh given 3D point clouds to create a library of generic plant templates, (ii) deform the selected plant template to fit the observation coming from a mobile robot to recover its 3D structure, and (iii) accurately measure leaf area on the deformed mesh targeting in-field applications. These claims are backed up by the paper and our experimental evaluation.

II. Related Works

Phenotyping using mobile robots is still limited to basic traits such as average plant height over the entire field. For example, the works by Carlone et al. [3] and by Chebrolu et al. [4], show point clouds of crop fields at different growth stages. Using non-rigid registration techniques they can estimate how the height of the plants changes by aligning the different point clouds. Additionally, a couple of works integrates prior knowledge of plant structures into 3D measurements. Binney et al. [1] fit cylinders to point clouds of trees to recover missing data. However, the most similar work to ours is Sodhi et al. [30], which addresses the problem of mapping plant sub-units called plant phytomers to their phenotype values involving sampling of parameterized 3D plant models from an underlying probability distribution, thus casting phenotyping as a search in the space of plant models. Our work is different in two aspects. First, we want to recover the 3D structure of the whole plant, not only its sub-units. Second, we use images as input instead of point clouds (we only use a 3D point cloud to compute plant templates when building the template database). Potentially, one could also define the template model using a computer graphics engine [6], thus removing entirely the need for capturing 3D data of plants.

The task of estimating 3D shapes of plants can be seen as a non-rigid registration problem. Non-rigid registration techniques can register scans with localized deformations in contrast to rigid registration techniques such as iterative closest point (ICP). One can divide these approaches into two categories. On one side, approaches that explicitly compute the data association between source and target point clouds [31], [42], [5]. On the other side, approaches that cast the registration task as a probability density estimation problem [27], [26]. In both cases, source and target are instances of 3D data. This is different in our case as we deal with heterogeneous inputs: we try to align a 3D source mesh to a target image. We do that in a way such that the rendered view of the mesh aligns with the image.

Differentiable rendering can deal with such heterogeneity. In a nutshell, it defines an approximation of the standard rasterization method such that it can be differentiated with respect to different parameters, i.e., materials, illuminations, camera poses, etc. [20]. In recent years, differentiable rendering modules have been used on top of neural network frameworks for a variety of vision tasks such as view

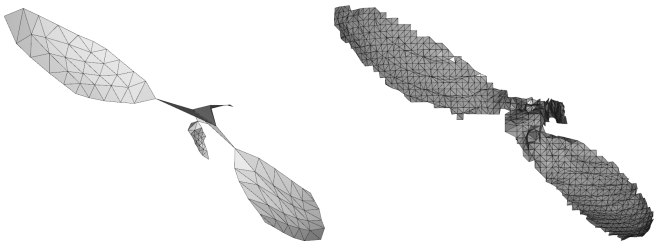


Fig. 3: An example of our 3D reconstruction (left) compared to the state-of-the-art (right) Poisson reconstruction [14]. While the captured topology is similar, our approach has fewer vertices. In this way, we simplify and speed up the optimization procedure.

synthesis [24], relighting [23], or material estimation [29]. For a complete overview of the subject, we refer to the state-of-the-art report by Tewari et al. [34]. The drawback of these approaches is the amount of data required for the training and the lack of generalization capability. Therefore, we follow an approach without a learning step and use the rendering definition of Kato et al. [13]. We couple it with ideas from the non-rigid registration domain, ensuring that the deformed model maintains the topology and aspect of a plant.

Additionally, in human motion analysis, a large amount of study deals with heterogeneous input where a human skeleton is deformed to fit the target image [38]. Skeleton fitting of humans is possible since the skeleton structure does not change and the difference in the target pose can be determined with a kinematic chain of the skeleton [37], [2], [33]. We take the idea of skeleton fitting and extend it to the agricultural scenario by computing a simplified mesh that we deform using a differentiable rendering approach.

III. Our Approach

Our approach takes as input single images from an on-board camera and a library of generic plant templates. In this paper, we use few highly accurate point clouds of plants obtained with a laser scanner from which we extract a simplified 3D mesh that we use as a template. Our goal is to deform the template to fit the current observation of the robot, thus enabling phenotypic measurements on the deformed mesh. See Fig. 2 for an overview of our approach. As a preprocessing step, we use the work of Weyler et al. [35] that provides pixel-wise instance segmentation of plants and an estimate of the BBCH index. i.e., its growth stage.

A. Mesh Extraction for Building a Template Database

To define a library of generic plant templates, we use few highly accurate 3D point clouds of plants. We compute a triangle mesh representation for these plant templates from the respective point clouds. This computation starts by classifying each point in the point cloud as stem or one leaf instance. This step is necessary as it will enable us to compute phenotypic traits afterward. Once the point cloud is segmented, we compute a grid structure for each organ, i.e., stem or leaves, based on self-organizing maps (SOM) [17] used in prior work [22].

SOMs are unsupervised neural networks using competitive learning instead of backpropagation. They take as input a grid that organizes itself to capture the topology of the input data. Given the input grid $\mathcal{G}$ and the input set $\mathcal{P}$, in our case both $\mathcal{G}$ and $\mathcal{P}$ are composed of points in $\mathbb{R}^3$, the SOM defines a fully-connected layer between $\mathcal{G}$ and $\mathcal{P}$. The learning process is composed of two alternating steps until convergence. First, the winning unit is computed as the $\mathrm{argmin}_i \, ||\mathbf{x}-\mathbf{w}_i||$, where $\mathbf{x}$ is a randomly chosen sample from $\mathcal{P}$ and $\mathbf{w}_i$ is the weight vector most similar to $\mathbf{x}$, also called best matching unit. The second step consists of updating the weights of each unit according to $\mathbf{w}_n = \mathbf{w}_n + \eta\,\beta(i)\,(\mathbf{x} - \mathbf{w}_i)$, where η is the learning rate and $\beta(i)$ a function, which weights the distance between unit n and the best matching unit. The SOM approach computes a simplified mesh with fewer vertices than the state-of-the-art method for 3D reconstruction such as Poisson [14]. The fewer number of vertices simplifies and accelerates the optimization procedure while capturing the geometry of the considered plant well, see Fig. 3 for an example.

B. Differentiable Rendering

Given a 3D mesh, such as those present in the template database, we can define a differentiable rendering operator to compare the 3D template to the current observation of the robot. With differentiable rendering, we want to generate images from a parametrized mesh that not only provides a rendered view of this mesh but also allows for differentiation with respect to the mesh vertices. In this way, we can determine how the mesh should be deformed to match the desired goal.

Using a differentiable renderer, we can define a loss function and use gradient descent to minimize such loss, exploiting modern machine learning frameworks to speed up the optimization. We use the definition of the 3D mesh renderer by Kato et al. [13]. We consider the value $\mathbf{v}$ of a pixel as a function $\mathfrak{r}$ of the mesh vertices, $P = \mathfrak{r}(\mathbf{v})$. Representing the displacement of a vertex $\mathbf{v}_i$ as $\delta_i^{\mathbf{v}} = \mathbf{v}_1 - \mathbf{v}_0$, where $\mathbf{v}_0$ and $\mathbf{v}_1$ represent its extremes, and its corresponding change in the rendered value $\delta_i^{\mathfrak{r}} = \mathfrak{r}(\mathbf{v}_1) - \mathfrak{r}(\mathbf{v}_0)$. In the standard rasterization method, the derivative $\frac{\partial \mathfrak{r}(\mathbf{v})}{\partial \mathbf{v}}$ is zero almost everywhere, thus there will be no gradient flow in the backpropagation step. This is due to the fact that the value of the pixel changes suddenly when the face that influences the rendered value changes. To solve this issue, the derivative $\frac{\partial \mathfrak{r}(\mathbf{v})}{\partial \mathbf{v}}$ becomes $\frac{\delta_i^{\mathfrak{r}}}{\delta_i^{\mathbf{v}}}$ between $\mathbf{v}_0$ and $\mathbf{v}_1$, representing a gradual change in the considered displacement. Once the differentiable renderer is defined, we can use it, paired with optimization algorithms developed in the context of deep learning, to deform the source mesh by minimizing the norm between its rendered view and the target image, such that rendered image will be as similar as possible to the target image.

C. Differentiable Rendering Meets Non-rigid Registration

We are not interested in the rendered image as most rendering systems. Instead, we are interested in the deformed

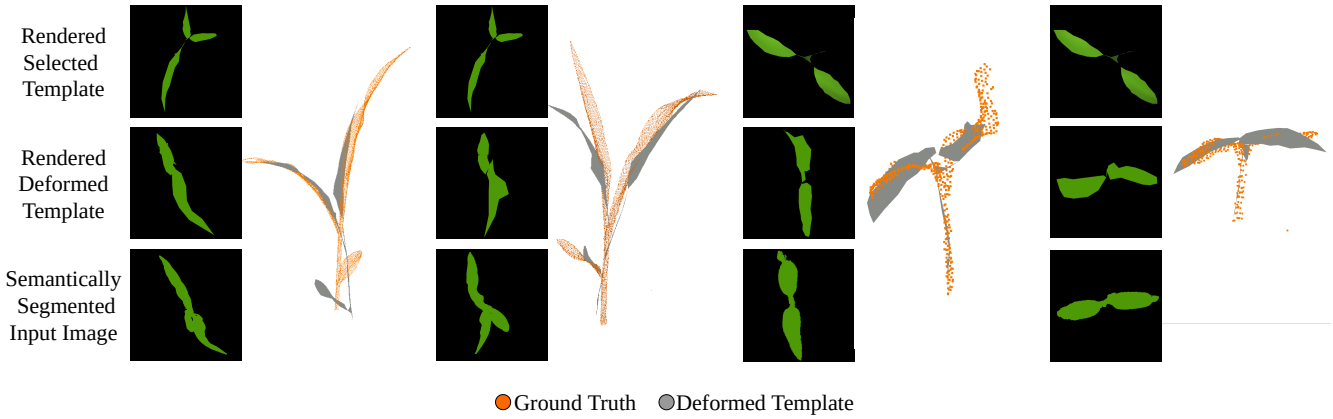


Fig. 4: Our approach can estimate the 3D geometry of a plant using single images. We overlay the deformed template to the original point cloud. On the side, we show the input image as well the rendered template before and after the optimization to appreciate the deformation results of our approach. (Best viewed in color.)

mesh that generates such an image. However, minimizing the norm between the rendered image and the target image gives no guarantee that the deformed mesh will have a meaningful topology. To overcome this issue during the optimization procedure, we integrate a loss function that tries to maintain the aspect of a plant. We define as $\mathcal{P}$ the plant template and with $\hat{\mathcal{P}}$ its deformed version, both meshes, $\mathcal{P}$ and $\hat{\mathcal{P}}$, are defined by a set of vertices $\mathcal{V} = \{\mathbf{v}_0, \mathbf{v}_1, ..., \mathbf{v}_n\}$ and a set of edges $\mathcal{E} = \{\mathbf{e}_0, \mathbf{e}_1, ..., \mathbf{e}_n\}$. We represent the target image as I and $\mathfrak{r}(\cdot)$ refers to the rendering function.

Inspired by the seminal works of Sorkine et al. [31] and Sumner et al. [32], we design a loss function that penalizes large displacement in vertices lying on the same edge of the template, i.e.,

$$\begin{aligned}\mathcal{L}(\mathcal{P}, \hat{\mathcal{P}}, \mathrm{I}) = \; & w_r ||\mathfrak{r}(\hat{\mathcal{P}}) - \mathrm{I}|| + w_f \sum_t ||\mathbf{n}_t - \hat{\mathbf{n}}_t|| \\ & + w_e \sum_{i,j \in \mathcal{E}} || \, |\mathbf{e}_{i,j}| - |\hat{\mathbf{e}}_{i,j}| \, || \\ & + w_d \sum_{i,j \in \mathcal{E}} ||\mathrm{dist}(\mathbf{v}_i) - \mathrm{dist}(\mathbf{v}_j)||,\end{aligned} \tag{1}$$

where $t = (i, j, k)$ is a triplet of indices defining a triangle with normal $\mathbf{n}_t = (\mathbf{v}_j - \mathbf{v}_i) \times (\mathbf{v}_k - \mathbf{v}_i)$ and $\mathrm{dist}(\mathbf{v}_i) = ||\hat{\mathbf{v}}_i - \mathbf{v}_i||$ is the displacement of vertex $\mathbf{v}_i$.

Intuitively, in Eq. (1), the first term of the loss function is the pixel-wise norm between the rendered mesh and the input image. The second and third terms penalize large deformation between the input template and the deformed template. The last term enforces similar deformations of points that are on the same edge. Note that there is no change in the mesh connectivity, and thus there is no need to compute the correspondences between input template and deformed template. Finally, we weigh each term by a different factor to obtain values in a similar order of magnitude.

D. Leaf Area Index as a Phenotypic Trait Extracted from the 3D Model

After the deformation of our plant template based on the current observation, we measure the leaf area by summing the area of the triangles t on the 3D model that contains at least one vertex classified as leaf. Dividing this value by the area of the projection of the same vertices on the ground plane, we can easily measure the leaf area index:

$$\mathrm{LAI} = \sum_t \frac{\mathrm{area}(\triangle_t)}{\mathrm{area}(\pi(\triangle_t))}, \tag{2}$$

where $\triangle_i$ is the i-th face in the template and $\pi(\triangle_i)$ is its projection on the ground plane.

IV. Experimental Evaluation

We define two experiments to showcase the capabilities of our approach. First, we compare the results of our registration procedure against a highly accurate 3D model obtained manually with the Romer Absolute Arm. Second, we can compute accurate phenotypic traits such as the leaf area index from a single image.

A. Dataset

To validate our approach, we manually record 3D point clouds of different species and different growth stages using a Romer Absolute Arm with a laser scanner as the end effector. Such a setup provides a sub-millimeter accuracy, thus we can use the obtained point clouds as ground truth to measure the accuracy of our differentiable rendering. In sum, we use 45 point clouds of tomato and 25 point clouds of maize plants. We use two point clouds for each species that we use as templates, while we use the rest for the experimental evaluation. We call these datasets tomato1, tomato2, maize1, maize2, where the different numbers refer to different growth stages. For each point cloud in the test set, we also render the corresponding image from the standard top-down view. These images form the input of our pipeline, while we use the 3D point cloud to compute the reconstruction accuracy.

B. Reconstruction Results

In the first experiment, we show that our approach can accurately recover 3D shapes given a single image and a plant template. In Fig. 4, we show example results from our

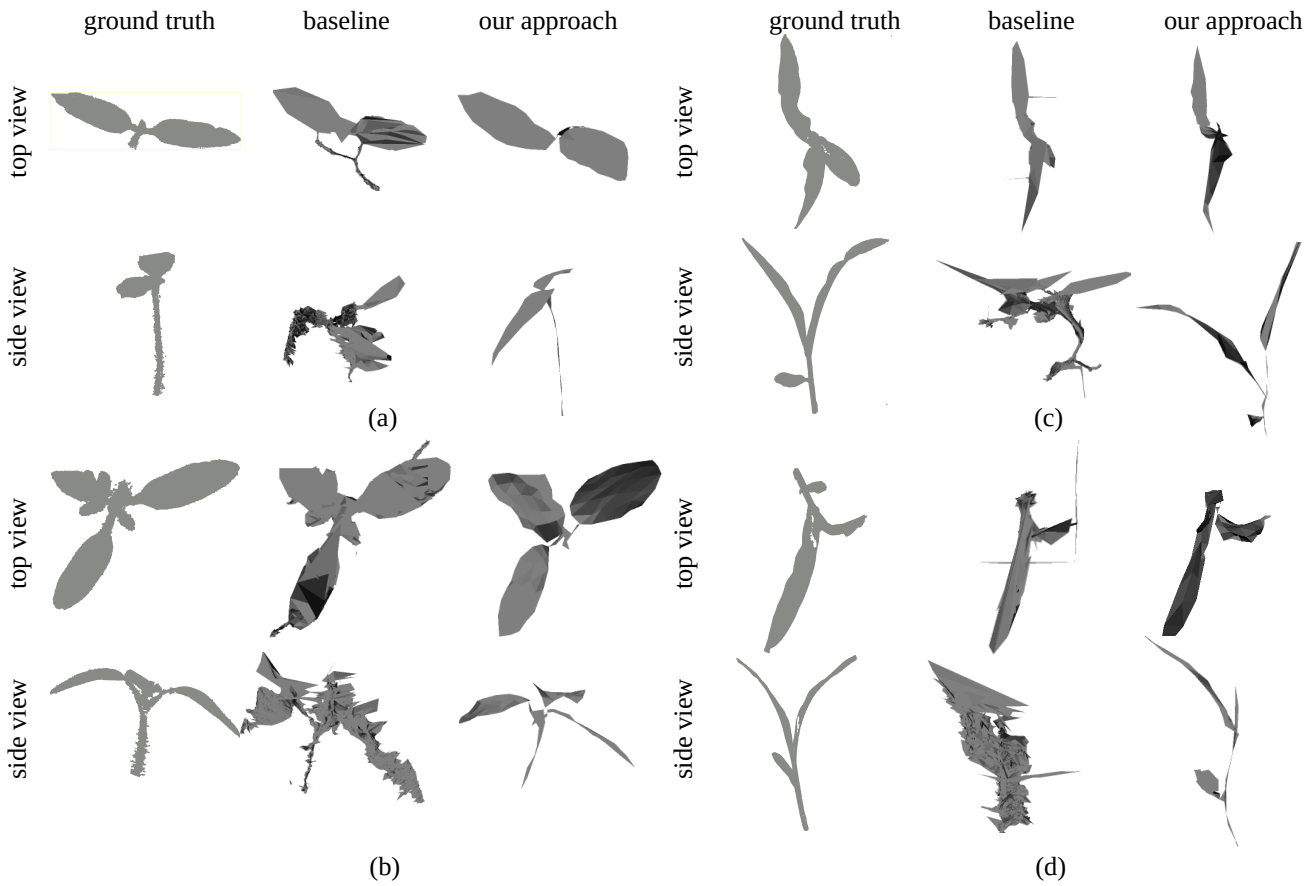


Fig. 5: Qualitative results of our approach. We show two examples of tomato plants, (a) and (b), and two examples of maize plants, (c) and (d). For each example, we present top and side views. The baseline correctly optimizes the top view, but the resulting 3D models lost the topology of a plant, instead, our approach can maintain the topology of a plant after deformation on occluded regions.

pipeline. For each example, we show the deformed mesh imposed over the ground truth (right) together with the target image and rendered template, before and after deformation (left). To measure the accuracy of our approach we use the f-score metric as described by Knapitsch et al. [16]. To compute the f-score, we first define precision p, and recall r, given a threshold δ:

$$
\begin{aligned}
p(\delta) &= \frac{100}{|\mathcal{R}|} \sum_{\boldsymbol{r} \in \mathcal{R}} \left[\!\!\left[\min_{\boldsymbol{g} \in \mathcal{G}} ||\boldsymbol{r} - \boldsymbol{g}|| < \delta \right]\!\!\right], \\
r(\delta) &= \frac{100}{|\mathcal{G}|} \sum_{\boldsymbol{g} \in \mathcal{G}} \left[\!\!\left[\min_{\boldsymbol{r} \in \mathcal{R}} ||\boldsymbol{g} - \boldsymbol{r}|| < \delta \right]\!\!\right],
\end{aligned}
\tag{3}
$$

where $\mathcal{G}$ and $\mathcal{R}$ are respectively the ground truth point cloud and the point cloud obtained by sampling the deformed template, $\boldsymbol{g}$ and $\boldsymbol{r}$ are points from $\mathcal{G}$ and $\mathcal{R}$ and the operator $[\![\cdot]\!]$ is the Iverson bracket, i.e., if the condition within the brackets is satisfied it evaluates to 1, otherwise to 0. Intuitively, such metrics compute the percentage of points in one set whose distance to the closest point in the other set is smaller than a fixed threshold. As always, the f-score is the harmonic mean of precision and recall $f(\delta) = \frac{2 \cdot p(\delta) \cdot r(\delta)}{p(\delta) + r(\delta)}$.

We compare our approach against the original work by Kato et al. [13] where the renderer does not enforce the self-consistency of the 3D models. In Fig. 6, we present the f-score results at different thresholds. Our approach yields better accuracy than the baseline, especially for lower thresholds, below 1 cm. We also show a qualitative comparison in Fig. 5, where we present top and side views for four examples of our results compared to the ground truth and the baseline. Our approach can maintain the plant topology after the deformation even in occluded regions thanks to our loss definition. For each sample in our dataset, we use the following weights in Eq. (1), $w_r = 0.01$, $w_e = 10$, $w_f = 100$, $w_d = 100$, and perform 1000 iterations using the Adam optimizer [15] with the learning rate $lr = 0.01$.

C. Ablation Study

To prove the importance of our choices, both the SOM-based meshing and the self-consistency loss, we perform an ablation study in which we try different combinations with state-of-the-art approaches in 3D reconstruction [14] and differentiable rendering [13]. We summarize the ablation study in Tab. I, where we indicate as $\mathcal{L}_r$ the rendering loss and as $\mathcal{L}_c$ our self-consistency loss. It is clear that both choices are important to achieve better reconstruction accuracies. In fact, using the SOM-based meshing without self-consistency loss does not guarantee any improvement

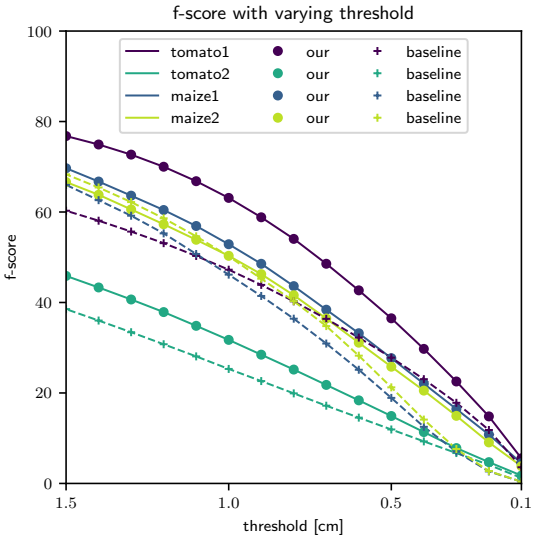


Fig. 6: We plot the f-score value for different thresholds for each of the datasets used in this paper. Our approach provides, in general, better results than the baseline, especially for lower thresholds. (Best viewed in color.)

TABLE I: Ablation study.

Approach		f-score, with $d = 1$cm [%]			
Template	Loss	maize1	maize2	tomato1	tomato2
Poisson	$\mathcal{L}_r$	46.15	**50.21**	47.24	25.31
Poisson	$\mathcal{L}_r + \mathcal{L}_c$	47.47	46.59	10.92	8.98
SOM	$\mathcal{L}_r$	45.01	45.48	34.18	30.38
SOM	$\mathcal{L}_r + \mathcal{L}_c$	**52.87**	**50.31**	**63.11**	**31.73**

compared to the baseline. The same applies to using our self-consistency loss to deform a 3D mesh obtained with the Poisson reconstruction. Instead, using SOM meshing and self-consistency loss we considerably improve the reconstruction results on our datasets.

D. Measuring LAI as a Phenotypic Trait

To show that our approach is useful for phenotypic applications, we compute the leaf area index (LAI) on the deformed template and we compare those measurements with the LAI that we manually measured from the plants in our dataset. Note that our approach is not specifically designed to compute the LAI, instead, other phenotypic traits (leaf length, stem diameter, etc.) can be computed as well. We present the phenotypic evaluation in Fig. 7, where on the x-axis we plot the LAI measured on the deformed template and on the y-axis the ground truth value of the LAI. We evaluate the Pearson's correlation index for all the datasets used in this paper, we obtain an average coefficient of 0.76 indicating a positive correlation between the computed area from the deformed template and the ground truth area.

V. Discussion

An intrinsic limitation of using a single top-down view is the lack of a penalty term considering occluded parts of

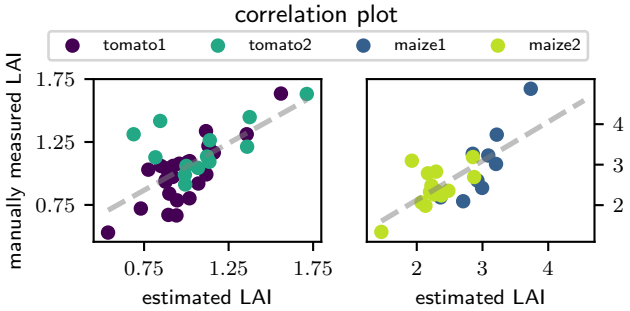


Fig. 7: Correlation plot between the leaf area extracted from the deformed template and the manually measured area from the ground truth point clouds. On average we obtain a correlation value of 0.76, indicating that with our approach, it is possible to get similar values for the leaf area using images instead of highly accurate 3D point clouds. (Best viewed in color.)

the canopy or displacements in the z-axis, see also Fig. 4, first two columns. A potential, yet simple, solution could be to use different images with different points of view in the optimization. This might be done by either optimizing for the different views at the same time or using one image after the other.

Additionally, the template database could become a bottleneck in an application using plants at later growth stages due to the increasing complexity of the plant topology. To tackle this issue, we see two directions. The first one is to rely on synthetically generated models [43], [6] instead of scanning real plants in a controlled environment. The second one is to define a parametrization of the template to relax the assumption of using the growth stage of plants explicitly. In this way, one could adapt the template itself by defining an optimization problem in the parameters space. Note that such directions are orthogonal to each other and can be applied together.

VI. Conclusions

In this paper, we presented a novel approach to perform advanced phenotypic measurements using single images. Our approach uses 2D images and is not bounded to a specific platform such that it can be used on robots, smartphones, and similar devices. Our method combines differentiable rendering with findings from non-rigid registration. This allows us to successfully deform a plant template in the form of a 3D mesh so that its rendered view fits the input image. As a result, we can measure phenotypic traits directly on the deformed template. We implemented and evaluated our approach on different plant species datasets at different growth stages, provided comparisons to other existing techniques, and supported all claims made in this paper. The experiments suggest that our approach maintains the plant geometry after the deformation, leading to accurate 3D reconstruction. Additionally, the phenotypic evaluation on the deformed template shows a positive correlation to the ground truth measurements.

REFERENCES

[1] J. Binney and G.S. Sukhatme. 3d tree reconstruction from laser range data. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, pages 1321–1326.

[2] L.W. Campbell and A.F. Bobick. Recognition of human body motion using phase space constraints. In *Proc. of the IEEE Intl. Conf. on Computer Vision (ICCV)*, pages 624–630, 1995.

[3] 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.

[4] N. Chebrolu, T. Läbe, 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.

[5] N. Chebrolu, T. Laebe, and C. Stachniss. Spatio-temporal non-rigid registration of 3d point clouds of plants. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[6] M.D. Cicco, C. Potena, G. Grisetti, and A. Pretto. Automatic Model Based Dataset Generation for Fast and Accurate Crop and Weeds Detection. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2017.

[7] T. Duckett, S. Pearson, S. Blackmore, B. Grieve, W. Chen, G. Cielniak, J. Cleaversmith, J. Dai, S. Davis, C. Fox, et al. Agricultural robotics: the future of robotic agriculture. *arXiv preprint arXiv:1806.06762*, 2018.

[8] F. Fiorani and U. Schurr. Future scenarios for plant phenotyping. *Annual Review of Plant Biology*, 64:267–291, 2013.

[9] J.A. Gibbs, M. Poundl, A.P. French, D.M. Wells, E. Murchie, and T. Pridmore. Approaches to three-dimensional reconstruction of plant shoot topology and geometry. *Functional Plant Biology*, 44(1):62–75, 2017.

[10] D. Gogoll, P. Lottes, J. Weyler, N. Petrinic, and C. Stachniss. Unsupervised Domain Adaptation for Transferring Plant Classification Systems to New Field Environments, Crops, and Robots. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[11] F. Golbach, G. Kootstra, S. Damjanovic, G. Otten, and van de Zedde R. Validation of plant part measurements using a 3d reconstruction method suitable for high-throughput seedling phenotyping. *Machine Vision and Applications*, 27(5):663–680, 2016.

[12] M. Hess, G. Barralis, H. Bleiholder, L. Buhr, T.H. Eggers, H. Hack, and R. Stauss. Use of the extended BBCH scale—general for the descriptions of the growth stages of mono; and dicotyledonous weed species. *Weed Research*, 37(6):433–441, 1997.

[13] H. Kato, Y. Ushiku, and T. Harada. Neural 3d mesh renderer. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[14] M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, volume 7, 2006.

[15] D.P. Kingma and J.Ba. Adam: A method for stochastic optimization. *arXiv preprint*, abs/1412.6980, 2014.

[16] A. Knapitsch, J. Park, Q. Zhou, and V. Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics*, 36(4):1–13, 2017.

[17] T. Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990.

[18] K. Kusumam, T. Krajník, S. Pearson, T. Duckett, and G. Cielniak. 3d-vision based detection, localization, and sizing of broccoli heads in the field. *Journal of Field Robotics (JFR)*, 34(8), 2017.

[19] C. Lehnert, D. Tsai, A. Eriksson, and C. McCool. 3d move to see: Multi-perspective visual servoing towards the next best view within unstructured and occluded environments. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2019.

[20] M.M. Loper and M.J. Black. Opendr: An approximate differentiable renderer. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, pages 154–169, 2014.

[21] P. Lottes, J. Behley, N. Chebrolu, A. Milioto, and C. Stachniss. Robust joint stem detection and crop-weed classification using image sequences for plant-specific treatment in precision farming. *Journal of Field Robotics (JFR)*, 37:20–34, 2020.

[22] F. Magistri, N. Chebrolu, and C. Stachniss. Segmentation-Based 4D Registration of Plants Point Clouds for Phenotyping. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[23] M. Meshry, D.B. Goldman, S. Khamis, H. Hoppe, R. Pandey, N. Snavely, and R. Martin-Brualla. Neural rerendering in the wild. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 6878–6887, 2019.

[24] B. Mildenhall, P.P. Srinivasan, M. Tancik, J.T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[25] A. Milioto, P. Lottes, and C. Stachniss. Real-time Semantic Segmentation of Crop and Weed for Precision Agriculture Robots Leveraging Background Knowledge in CNNs. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2018.

[26] Z. Min, J. Pan, A. Zhang, and M. Q. H. Meng. Robust non-rigid point set registration algorithm considering anisotropic uncertainties based on coherent point drift. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 7903–7910, 2019.

[27] A. Myronenko and X. Song. Point set registration: Coherent point drift. *IEEE Trans. on Pattern Analalysis and Machine Intelligence (TPAMI)*, 32(12):2262–2275, 2010.

[28] P. Prusinkiewicz and A. Lindenmayer. *The algorithmic beauty of plants*. Springer Science & Business Media, 2012.

[29] S. Sengupta, J. Gu, K. Kim, G. Liu, D.W. Jacobs, and J. Kautz. Neural inverse rendering of an indoor scene from a single image. In *Proc. of the IEEE Intl. Conf. on Computer Vision (ICCV)*, pages 8598–8607, 2019.

[30] P. Sodhi, H. Sun, B. Póczos, and D. Wettergreen. Robust plant phenotyping via model-based optimization. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 7689–7696, 2018.

[31] O. Sorkine and M. Alexa. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, volume 4, pages 109–116, 2007.

[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, M. Schröder, A. Tkach, S. Bouaziz, M. Botsch, and M. Pauly. Robust articulated-icp for real-time hand tracking. volume 34, pages 101–114, 2015.

[34] A. Tewari, O. Fried, J. Thies, V. Sitzmann, S. Lombardi, K. Sunkavalli, R. Martin-Brualla, T. Simon, J. Saragih, M. Nießner, et al. State of the art on neural rendering. *Eurographics - State-of-the-Art Reports (STARs)*, 2020.

[35] J. Weyler, A. Milioto, T. Falck, J. Behley, and C. Stachniss. Joint Plant Instance Detection and Leaf Count Estimation for In-Field Plant Phenotyping. *IEEE Robotics and Automation Letters (RA-L)*, 2021.

[36] X. Wu, S. Aravecchia, P. Lottes, C. Stachniss, and C. Pradalier. Robotic weed control using automated weed and crop classification. *Journal of Field Robotics (JFR)*, 37:322–340, 2020.

[37] L. Xia, C. Chen, and J. Aggarwal. View invariant human action recognition using histograms of 3d joints. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 20–27, 2012.

[38] M. Ye, Q. Zhang, L. Wang, J. Zhu, R. Yang, and J. Gall. A survey on human motion analysis from depth data. In *Time-of-flight and depth imaging. sensors, algorithms, and applications*. Springer, 2013.

[39] L. Zabawa, A. Kicherer, L. Klingbeil, R. Töpfer, H. Kuhlmann, and R. Roscher. Counting of grapevine berries in images via semantic segmentation using convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing (JPRS)*, 164:73–83, 2020.

[40] D. Zermas, V. Morellas, D. Mulla, and N. Papanikolopoulos. Estimating the leaf area index of crops through the evaluation of 3d models. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 6155–6162, 2017.

[41] D. Zermas, V. Morellas, D. Mulla, and N. Papanikolopoulos. Extracting phenotypic characteristics of corn crops through the use of reconstructed 3d models. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 8247–8254, 2018.

[42] 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.

[43] X-R. Zhou, A. Schnepf, J. Vanderborght, D. Leitner, A. Lacointe, H. Vereecken, and G. Lobet. Cplantbox, a whole-plant modelling framework for the simulation of water-and carbon-related processes. *in silico Plants*, 2(1), 2020.