# Panoptic Mapping with Fruit Completion and Pose Estimation for Horticultural Robots

Yue Pan     Federico Magistri     Thomas Läbe     Elias Marks
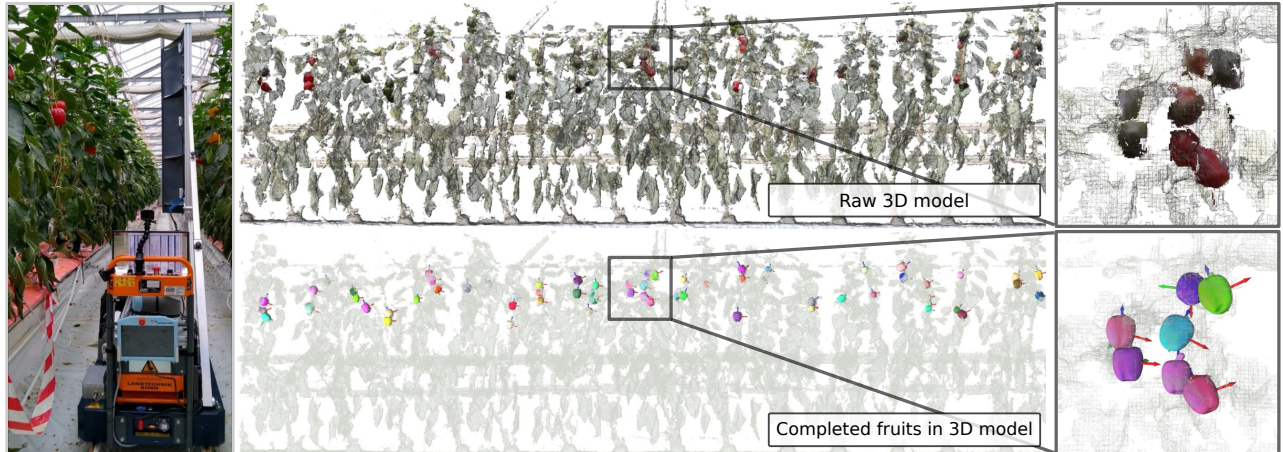Claus Smitt     Chris McCool     Jens Behley     Cyrill Stachniss

Fig. 1: Our method is able to build a multi-resolution panoptic map (top) of a challenging commercial greenhouse environment online using a mobile horticultural robot equipped with RGB-D cameras (left). Furthermore, our method manages to jointly estimate the complete shape and pose of each fruit in the panoptic map (bottom).

*Abstract*—**Monitoring plants and fruits at high resolution play a key role in the future of agriculture. Accurate 3D information can pave the way to a diverse number of robotic applications in agriculture ranging from autonomous harvesting to precise yield estimation. Obtaining such 3D information is non-trivial as agricultural environments are often repetitive and cluttered, and one has to account for the partial observability of fruit and plants. In this paper, we address the problem of jointly estimating complete 3D shapes of fruit and their pose in a 3D multi-resolution map built by a mobile robot. To this end, we propose an online multi-resolution panoptic mapping system where regions of interest are represented with a higher resolution. We exploit data to learn a general fruit shape representation that we use at inference time together with an occlusion-aware differentiable rendering pipeline to complete partial fruit observations and estimate the 7 DoF pose of each fruit in the map. The experiments presented in this paper, evaluated both in the controlled environment and in a commercial greenhouse, show that our novel algorithm yields higher completion and pose estimation accuracy than existing methods, with an improvement of 41% in completion accuracy and 52% in pose estimation accuracy while keeping a low inference time of 0.6 s in average.**

## I. INTRODUCTION

To feed an ever-growing world population, the whole agricultural sector needs to increase its productivity while re-

ducing its negative impact on the environment. Autonomous robots have the potential to support addressing both issues. For example, robots can automate labor-intensive tasks such as harvesting [2], [11], weeding [18], or pruning [3]. They can also provide plant-specific treatment of herbicides and pesticides [1] reducing the required amount of agrochemicals. Thus increasing the likelihood of meeting the population demands in producing food, feed, fiber, and fuel and at the same time decreasing the use of agro-chemicals.

Robots can continuously monitor orchards or arable fields to detect early stages of plant stress [34], support phenotyping activities [36], and provide detailed yield estimates [9]. Robots working in arable fields or horticulture environments can seldomly observe the whole scene due to the cluttered nature of the environments. This means that data obtained with any agricultural robot is partial and incomplete. In this paper, we address the problem of building multi-resolution 3D maps of such scenes and estimating the non-visible parts of fruit to obtain more complete 3D models. Fig. 1 depicts an example of the resulting map together with the 3D shape and pose estimated for each fruit.

To recover the shapes of occluded objects, a typical solution is the usage of templates whose deformations are able to represent different instances of the same category. A template can be represented as a 3D triangular mesh [14] or encoded in the weights of a neural network [24]. In recent years, templates have been used to recover shapes of human bodies [4] and hands [31], but also in the agricultural context to estimate shapes of fruits [15] and plants [17].
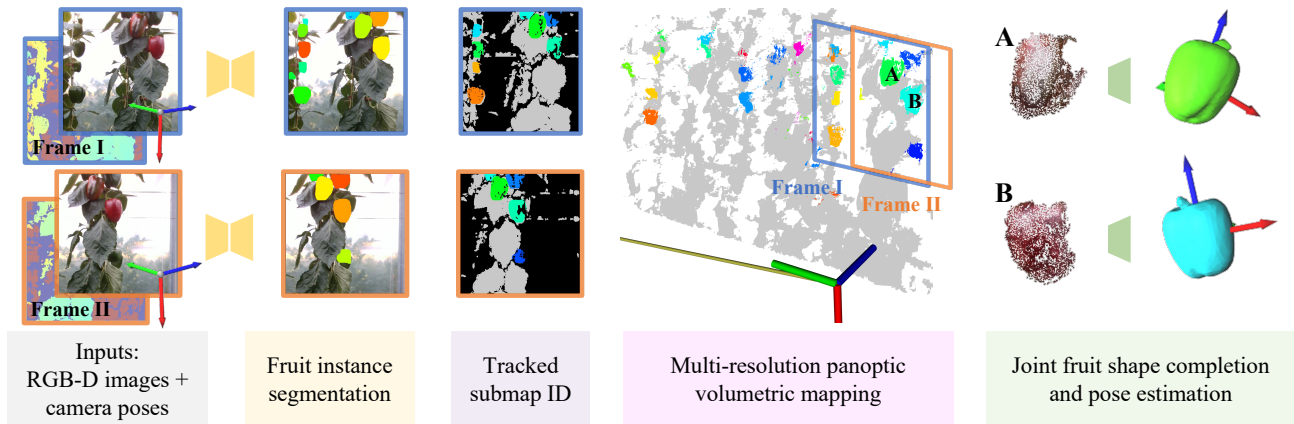
Fig. 2: Overview pipeline of the proposed mapping system: the input to the mapping system is a stream of the RGB-D images collected by the horticultural mobile robot. We predict the fruit instance using Mask R-CNN [8]. Then we conduct panoptic volumetric mapping by tracking the instance lables as the temporal consistent submap IDs. For each fruit submap, we jointly estimate the complete shape and pose from the submap points and the corresponding images using offline learned shape prior and the differentiable rendering pipeline.

The main contribution of this paper is a novel method to jointly estimate the 3D shape of fruits and their pose. We build multi-resolution maps in which we place the predicted 3D shapes of fruits correctly posed in a globally consistent representation. Additionally, we exploit high-resolution 3D data to encode a general fruit representation into the weights of a neural network. In this way, we can recover details of the fruit's shape with a low inference time during operations.

In sum, we make three key claims: our approach is able to jointly (i) predict the 3D shape of fruits even under substantial occlusions in real commercial greenhouse environments and (ii) estimate the pose of each fruit in the 3D map. (iii) Additionally, our multi-resolution map representation yields substantial improvement in the shape completion task over fixed-resolution maps. These claims are backed up by our experimental evaluation.

Our open-source implementation is publicly available at: https://github.com/PRBonn/HortiMapping.

## II. RELATED WORK

In recent years, agricultural robotics has become an increasingly popular research area due to the aforementioned challenges and the prospect of deploying such robots for more efficient and sustainable agri- and horticulture. More specifically, in the horticulture context we have seen deployed robotic systems for monitoring [30] and harvesting [2], [11]. In both cases, a fundamental build block of such robots is an instance segmentation network [8] that can robustly segment fruits [6], [19], peduncles [27], and plants [5], often starting from 2D images. Neural networks can also be used to estimate ripeness [7] and poses [32] of fruits. Based on such networks, it is possible to build semantically-aware 3D maps [25]. However, in the agricultural context, classical volumetric mapping pipelines, such as KinectFusion [21] or Voxblox [22], yield incomplete fruits representations as they do not deal explicitly with the presented occlusions. Instead, we build upon a multi-resolution map from prior work [23], [28] and propose a pipeline to jointly estimate the pose and

the complete shape of fruits in the 3D map resulting in a complete and better representation of fruits.

Recently, a variety of works tackle the problem of estimating the shape of non-visible fruits parts of plants or fruits. Such works can be generally divided into three categories: geometry-based, mesh-based, and deep learning-based approaches. In the first category, a closed-form geometric model is fit into the collected data. Marangoz et al. [16] use such an approach for fruit monitoring, while Lehnert et al. [12] show that fitting a superellipsoid improves robotic grasping performances. These approaches estimate complete shapes quickly but the model used cannot represent details in the 3D shapes.

In the second category, the model is represented by a 3D mesh that represents the general appearance of a target object. We used this approach to estimate the shapes of plants [14] and leaves [17]. While such methods can provide precise reconstructions, a shortcoming is the high inference time required for obtaining the final mesh. This is often not practical for real-world operations.

In the last category, the general appearance of an object is encoded in the weights of a neural network that produces either point clouds [35] or triangular meshes [24]. In our previous work [15], we use DeepSDF [24] to learn a prior over fruits shapes and map a single RGB-D frame to the network's latent space to avoid the online optimization. While such approaches provide plausible 3D shapes at a low inference time, they lack the ability to estimate 7 degrees of freedom (DoF) poses and the performance declines when processing the fruits with non-canonical poses in the real world. The effectiveness of DeepSDF-based methods [15] on in-the-wild settings is thus limited by the requirement of having the input partial shapes in the same canonical pose and scale as in the training set.

Inspired by the recent works in the computer vision community including Frodo [26] and DSP-SLAM [33] we propose a novel method to jointly estimate the shape and the pose of fruits with a low inference time. We combine a completion network based on a neural shape prior and

an occlusion-aware differentiable rendering pipeline. In this way, the shape completion and reconstruction are still conducted in the canonical pose of the fruit with high quality while obtaining the transformation to the world coordinate system and allowing for its embedding in a 3D model of the greenhouse and its plants.

## III. OUR APPROACH TO FRUIT MAPPING

In this paper, we study the problem of panoptic volumetric mapping for horticultural applications while estimating complete 3D shapes including scale and 7 DoF pose of fruits from a sequence of RGB-D frames.

As shown in Fig. 2, we build a panoptic volumetric map that decomposes the scene into the background and individual foreground submaps. The background submap is represented at a lower resolution than the foreground submaps to ensure accurate and high-fidelity reconstructions of the fruits. Based on the foreground submaps integrating partial observations, we estimate complete shapes and 7 DoF poses of each individual fruit using a deep neural network that jointly estimates the shape of the fruit in a canonical coordinate system and a transformation into the world coordinate frame.

### A. Multi-resolution Panoptic Mapping

The input to our mapping system is a stream of the RGB-D images collected by a horticultural mobile robot. The pose of the robot can be estimated by wheel odometry or a tracking camera and refined online with a standard RGB-D odometry system. We train a Mask R-CNN [8] model to perform online instance segmentation of the fruits based on the RGB information resulting in per-fruit instance masks $\mathcal{M}$. In line with Panoptic Multi-TSDFs [28], we assign a submap $\mathcal{S}$ to each panoptic entity, either the thing (fruit) instance or the stuff (background). For each submap, we apply a non-projective TSDF integration [23] as well as a mesh reconstruction based on marching cubes algorithm [13] incrementally using the RGB-D image stream and the estimated robot's poses.

Meanwhile, we track the segmented instances as temporal consistent submap IDs through a data association. We project the mesh vertices of active submaps within the visual frustum onto the image planes to generate submap masks for newly collected image frames. Based on the intersection over union (IoU) ratio between the rendered and predicted masks and the difference between the rendered and measured depth, we either assign a segmented instance to an existing submap or allocate a new submap for it.

As shown in Fig. 3, our volumetric mapping allows the fruits to be reconstructed at a small voxel size (3 mm) and with high accuracy, while using larger voxel sizes (1 cm) to efficiently map the background or less relevant classes such as leaves and stems. Therefore, the final panoptic map consists of a low-resolution submap for the background and high-resolution submaps for each fruit instance.

Once a fruit submap is not observed for $G$ frames consecutively, it is not updated anymore, marked as frozen, and used for joint shape completion and pose estimation.
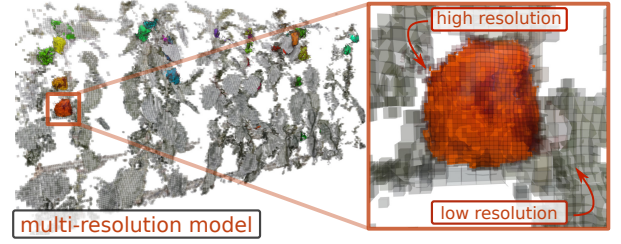


Fig. 3: On the left, we show an example of our multi-resolution volumetric map consisting of a lower resolution background (with larger voxel size) and several higher resolution fruit submaps (with smaller voxel size). On the right, we zoom in to show the high resolution submap representing a fruit.
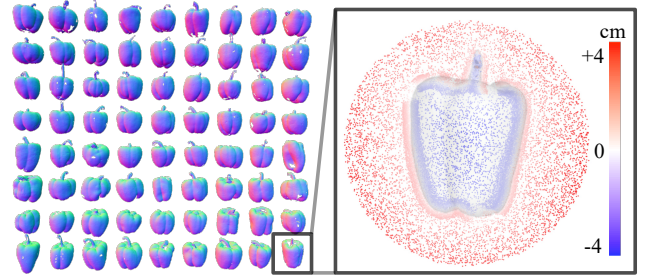


Fig. 4: On the left, we show examples of the sweet pepper models with a canonical pose measured using a sub-millimeter precision laser scanner. On the right, for one example sweet pepper, we show both the close-to-surface sample points and the uniform free space sample points with their color representing the SDF value label. We use these sample points and their SDF value labels to train the DeepSDF model and learn the shape prior of the specific fruit species (in this case the sweet pepper).

### B. Fruit Completion and Pose Estimation in the Wild

By exploiting the fruit shape prior of a pre-trained DeepSDF [24] model and an occlusion-aware differentiable rendering technique, for each frozen fruit submap $\mathcal{S}$ from the panoptic volumetric map in Sec. III-A, we aim to jointly estimate its latent shape code $z$ and 7 DoF transformation $\mathsf{T}_{ow} \in \mathrm{Sim}(3)$ from the world coordinate system to the fruit's canonical coordinate system. $\mathsf{T}_{ow}$ is represented by an homogeneous transformation matrix with $s \in \mathbb{R}$, $R \in \mathbb{R}^{3\times3}$, $t \in \mathbb{R}^3$ representing the scale, rotation, and translation, respectively. We sample a point cloud $\mathcal{P}_{\mathcal{S}}$ from the mesh of each instance submap $\mathcal{S}$. Additionally, from all the images in which $\mathcal{S}$ is visible, we get the 2D mask $\mathcal{M}$ predicted by Mask R-CNN, the extended 2D bounding box $\mathcal{B}$ with a padding of $h$ pixels on each side.

**DeepSDF pretraining:** The DeepSDF [24] model takes as input a query position $x \in \mathbb{R}^3$ and a latent shape code $z \in \mathbb{R}^C$, and predicts the SDF value $v \in \mathbb{R}$ at $x$ through a decoder network $D_\theta$ as: $v = D_\theta(x, z)$. With a pretrained DeepSDF decoder $D_\theta$ and optimized shape code $z$, we compute a dense SDF volume by querying it at a regular 3D grid of points, which we use for a complete mesh reconstruction via marching cubes [13].

In line with Magistri et al. [15], we aim at having an accurate and complete model of the fruits predicted by $D_\theta$ for which we use high-resolution 3D laser scans collected in
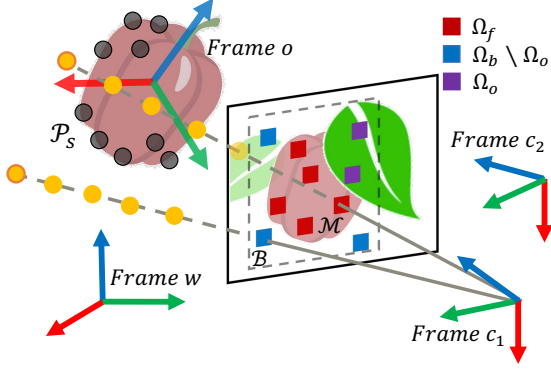
Fig. 5: Geometric relationships for the joint optimization of the shape $\boldsymbol{z}$ and the pose $\mathsf{T}_{ow}$ of a target fruit using the submap point cloud in the world frame $w$ and the image collected in the camera frame $c$. The black points represent the sampled points from the point cloud $\mathcal{P}_S$ of the target fruit submap. The orange points represent the sample points on each ray corresponding to each sampled pixel. The red, blue, and purple squares represent the sampled pixels from the masked foreground region $\mathcal{M}$, the unmasked background region in the extended bounding box $\mathcal{B}$ and the potential occluded background region, respectively.

a controlled laboratory environment using a sub-millimeter accurate Perceptron V5 laser scanner and a Romer Infinite measuring arm. We train a DeepSDF model for each type of fruit using the measured 3D scans and instead of only sampling points close to the surface along the normal direction [24], [15], we also sample points uniformly in a sphere surrounding the object as shown in Fig. 4. By doing so, the model better represents the free space, which is beneficial for differentiable rendering and pose estimation. Regarding model training, we adopt a latent code size of $C = 32$ and adhere to the network architecture and hyperparameter configurations specified in the original work of DeepSDF [24].

**Surface consistency loss:** Our first objective is to achieve precise alignment between the reconstructed fruit and the fused 3D observations from the RGB-D sensor. To accomplish this, as shown in Fig. 5, we aim to keep the points from the target fruit submap's point cloud $\mathcal{P}_S$ close to the iso-surface of the SDF predicted by $D_\theta$, which minimizes the following surface reconstruction loss $\mathcal{L}_s$ given by:

$$\mathcal{L}_s = \frac{1}{|\mathcal{P}_S|} \sum_{\mathbf{p}^w \in \mathcal{P}_S} D_\theta^2 \left(\mathsf{T}_{ow}\mathbf{p}^w, \boldsymbol{z}\right). \tag{1}$$

**Occlusion-aware differentiable rendering:** As we use depth images and instance segments, we additionally propose a depth rendering loss and a mask rendering loss using differentiable SDF rendering. As shown in Fig. 5, for each image corresponding to a visible fruit submap, we sample $G_f$ foreground pixels $\boldsymbol{\Omega}_f$ from the mask $\mathcal{M}$ and $G_b$ background pixels $\boldsymbol{\Omega}_b$ from the unmasked region $\mathcal{B} \setminus \mathcal{M}$ in the extended bounding box $\mathcal{B}$. For each sampled pixel $\boldsymbol{u} \in \boldsymbol{\Omega} = \boldsymbol{\Omega}_f \cup \boldsymbol{\Omega}_b$, we calculate the corresponding ray $\mathbf{r}$ in the camera coordinate system. We then sample $N + 1$ points with a fixed interval $\Delta d$ on each ray, resulting in per sample depths $d_i$:

$$d_i = d_{\min} + (i - 1)\Delta d, \qquad i = 1, \ldots, N+1, \tag{2}$$

along the ray $\mathbf{r}$ from the projection center to the fruit, where $d_{\min}$ and $\Delta d$ are determined by the distance from the camera to the submap's bounding box center and the approximate fruit size. By transforming each sample point to the world coordinate system, we obtain:

$$\mathbf{p}_i^w = \mathsf{T}_{wc}\left(d_i \mathsf{K}^{-1}\mathbf{u}\right), \tag{3}$$

where $\mathsf{K}$ and $\mathsf{T}_{wc}$ are the camera intrinsic and extrinsic matrices, respectively. Then, the SDF prediction $v_i$ at the sample point $\mathbf{p}_i^w$ is given by:

$$v_i = D_\theta(\mathsf{T}_{ow}\mathbf{p}_i^w, \boldsymbol{z}), \tag{4}$$

which can then be converted to an estimated occupancy probability $o_i$ by a logistic function with a surface noise threshold $\sigma$:

$$o_i = \frac{1}{1 + e^{v_i/\sigma}}. \tag{5}$$

Then, the so-called ray termination weight $\alpha_i$ for each sample point can be calculated from the occupancy probability as:

$$\alpha_i = \begin{cases} o_i \prod_{j=1}^{i-1}(1 - o_j) & i = 1, \ldots, N \\ \prod_{j=1}^{N}(1 - o_j) & i = N + 1. \end{cases} \tag{6}$$

Note that $\sum_{i=1}^{N+1} \alpha_i = 1$ always holds. We can then get the rendered mask $\hat{M}$ and depth $\hat{D}$ by integrating over all the samples along the ray as:

$$\hat{M} = \sum_{i=1}^{N} \alpha_i, \quad \hat{D} = \sum_{i=1}^{N+1} \alpha_i d_i. \tag{7}$$

The depth rendering loss $\mathcal{L}_d$ and mask rendering loss $\mathcal{L}_m$ for $K$ image frames observing the fruit submap are calculated by comparing the rendered result $\hat{D}$ and $\hat{M}$ with the depth camera's measurement $D$ and the binary mask $M$ of the target fruit for each of the $G$ sampled pixels as:

$$\mathcal{L}_d = \frac{1}{KG} \sum_k \sum_{\mathbf{u} \in (\boldsymbol{\Omega}^k \setminus \boldsymbol{\Omega}_o^k)} \left(\hat{D}_\mathbf{u} - D_\mathbf{u}\right)^2, \tag{8}$$

$$\mathcal{L}_m = \frac{1}{KG} \sum_k \sum_{\mathbf{u} \in (\boldsymbol{\Omega}^k \setminus \boldsymbol{\Omega}_o^k)} \left(\hat{M}_\mathbf{u} - M_\mathbf{u}\right)^2. \tag{9}$$

For background pixels, if $\hat{D}_\mathbf{u} - D_\mathbf{u} > d_o$, where $d_o$ is a small threshold, the pixel is regarded as lying in a potential occluded region of the fruit caused by leaves or other fruits, as shown in Fig. 5. Such occluded pixels $\boldsymbol{\Omega}_o \subset \boldsymbol{\Omega}_b$ are not taken into account in the rendering loss $\mathcal{L}_d$ and $\mathcal{L}_m$. For the rest of background pixels $\boldsymbol{\Omega}_b \setminus \boldsymbol{\Omega}_o$, we take the termination depth $d_{\max} = d_{\min} + N\Delta d$ as the virtual depth measurements for depth rendering loss calculation so that we can enforce the silhouette consistency at the target fruit.

### C. Optimization

With an additional shape code regularization term $\mathcal{L}_r = \|\boldsymbol{z}\|^2$, we use as our final loss function:

$$\mathcal{L} = w_s\mathcal{L}_s + w_d\mathcal{L}_d + w_m\mathcal{L}_m + w_r\mathcal{L}_r, \tag{10}$$

where $w_s$, $w_d$, $w_m$, $w_r$ are the weights for each loss term.

Our goal is to solve $\boldsymbol{\xi}_{ow}^*, \boldsymbol{z}^* = \operatorname{argmin} \mathcal{L}$, where $\boldsymbol{\xi}_{ow} \in \mathbb{R}^7$ is the corresponding $\mathfrak{sim}(3)$ Lie Algebra of $\mathsf{T}_{ow}$. Instead of using first-order optimization such as gradient descent, we use Levenberg-Marquardt with analytical Jacobians for faster and more stable convergence. The latent shape code $\boldsymbol{z}$ is initialized as $\boldsymbol{0}_C$ while $\boldsymbol{\xi}_{ow}$ is initialized as an identity rotation, scaling of 1, and a translation from the bounding box center of the submap point cloud to the origin. For each iteration, with a damping parameter $\lambda$, the increment $\delta\boldsymbol{x}$ to the estimated parameter vector $[\boldsymbol{\xi}_{ow}, \boldsymbol{z}]^\mathsf{T}$ is given by:

$$\delta\boldsymbol{x} = [\delta\boldsymbol{\xi}_{ow}, \delta\boldsymbol{z}]^\mathsf{T} = (H + \lambda \operatorname{diag}(H))^{-1} \boldsymbol{g} \qquad (11)$$

The approximate Hessian matrix is given by $H = J^\mathsf{T} P J$ and the gradient of the target function is $\boldsymbol{g} = J^\mathsf{T} P \boldsymbol{b}$, where $J$, $P$, $\boldsymbol{b}$ are the Jacobian matrix, weight matrix and residual vector, respectively.

Since both, the submap point cloud and the camera depth measurements are noisy, we apply a Huber robust kernel for the surface reconstruction and depth rendering residual resulting in the weight $w_i^h$ for each observation:

$$w_i^h = \begin{cases} 1 & \text{, if } |e_i| \le \tau \\ \tau/|e_i| & \text{, otherwise} \end{cases}, \qquad (12)$$

where $e_i$ and $\tau$ are the corresponding residual and kernel threshold, respectively. The weight matrix is formulated as $P = \operatorname{diag}(w_s w_1^h, \cdots, w_d w_n^h)$ with both the loss specific weight $w_s$, $w_d$ and the Huber weight $w^h$.

The residual and Jacobian of the surface reconstruction term for each submap point is given by:

$$\boldsymbol{b}_s = -D_\theta(\boldsymbol{p}^o, \boldsymbol{z}), \qquad J_s = \frac{\partial D_\theta(\boldsymbol{p}^o, \boldsymbol{z})}{\partial[\boldsymbol{\xi}_{ow}, \boldsymbol{z}]^\mathsf{T}}. \qquad (13)$$

Applying the chain rule, we obtain the derivative of $\boldsymbol{\xi}_{ow}$:

$$\frac{\partial D_\theta(\boldsymbol{p}^o, \boldsymbol{z})}{\partial\boldsymbol{\xi}_{ow}} = \frac{\partial D_\theta(\boldsymbol{p}^o, \boldsymbol{z})}{\partial\boldsymbol{p}^o} \frac{\partial\boldsymbol{p}^o}{\partial\boldsymbol{\xi}_{ow}}, \qquad (14)$$

$$\frac{\partial\boldsymbol{p}^o}{\partial\boldsymbol{\xi}_{ow}} = \frac{\partial(\mathsf{T}_{ow}\mathbf{p}^w)}{\partial\boldsymbol{\xi}_{ow}} = \begin{bmatrix} l_{3\times3} & -\boldsymbol{p}_\times^o & \boldsymbol{p}^o \end{bmatrix}, \qquad (15)$$

where $\times$ refers to the skew symmetric matrix of a vector. Note that both, $\frac{\partial D_\theta(\boldsymbol{p}^o, \boldsymbol{z})}{\partial\boldsymbol{p}^o}$ and $\frac{\partial D_\theta(\boldsymbol{p}^o, \boldsymbol{z})}{\partial\boldsymbol{z}}$ can be obtained through automatic differentiation of the DeepSDF model $D_\theta$. The residuals of the rendering term for each sampled pixel are simply:

$$\boldsymbol{b}_d = \hat{D} - D, \qquad \boldsymbol{b}_m = \hat{M} - M, \qquad (16)$$

and the Jacobians can also be obtained using the chain rule:

$$J_d = \sum_{i=1}^N \frac{\partial\hat{D}}{\partial o_i} \frac{\partial o_i}{\partial v_i} \frac{\partial D_\theta(\boldsymbol{p}_i^o, \boldsymbol{z})}{\partial[\boldsymbol{\xi}_{ow}, \boldsymbol{z}]^\mathsf{T}}, \qquad (17)$$

$$J_m = \sum_{i=1}^N \frac{\partial\hat{M}}{\partial o_i} \frac{\partial o_i}{\partial v_i} \frac{\partial D_\theta(\boldsymbol{p}_i^o, \boldsymbol{z})}{\partial[\boldsymbol{\xi}_{ow}, \boldsymbol{z}]^\mathsf{T}}, \qquad (18)$$
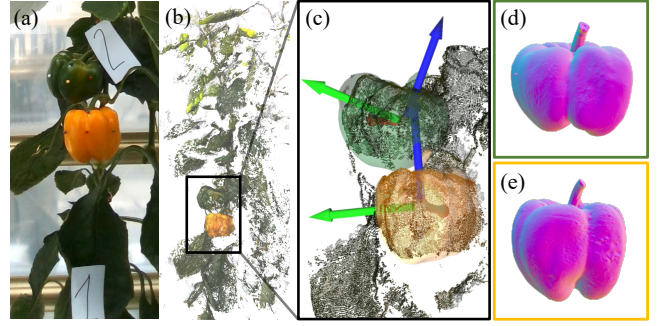


Fig. 6: Example from our greenhouse dataset: (a) shows the image of two sweetpeppers with the ID #1 and #2. (b) shows the point cloud generated from the camera of the robot by offline bundle adjustment. (c) shows the ground truth complete shape models with the ground truth pose manually aligned with the bundle adjusted point cloud. (d) and (e) show the ground truth shape models of sweetpepper #2 and #1 obtained by harvesting the fruits and scanning them in the laboratory with a high-precision laser scanner.

in which $\frac{\partial D_\theta(\boldsymbol{p}_i^o, \boldsymbol{z})}{\partial[\boldsymbol{\xi}_{ow}, \boldsymbol{z}]^\mathsf{T}}$ is in the same form as $J_s$ and the other three derivative terms are derived as:

$$\frac{\partial\hat{D}}{\partial o_i} = \frac{\Delta d}{1 - o_i} \sum_{j=i}^N \prod_{k=1}^j (1 - o_k), \qquad (19)$$

$$\frac{\partial\hat{M}}{\partial o_i} = \prod_{j=1, j\neq i}^N (1 - o_j), \qquad (20)$$

$$\frac{\partial o_i}{\partial v_i} = \frac{o_i(1 - o_i)}{\sigma}. \qquad (21)$$

Lastly, the residual and Jacobian of the shape code regularization term are:

$$\boldsymbol{b}_r = -\boldsymbol{z} \qquad (22)$$

$$J_r = \begin{bmatrix} \frac{\partial\boldsymbol{z}}{\partial\boldsymbol{\xi}_{ow}} & \frac{\partial\boldsymbol{z}}{\partial\boldsymbol{z}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{0}_{1\times7} & \boldsymbol{1}_{1\times C} \end{bmatrix} \qquad (23)$$

With all Jacobians and residuals available, we are able to solve Eq. (11) and update:

$$\begin{bmatrix} \boldsymbol{\xi}_{ow}^{(t+1)}, \boldsymbol{z}^{(t+1)} \end{bmatrix}^\mathsf{T} = \begin{bmatrix} \boldsymbol{\xi}_{ow}^{(t)}, \boldsymbol{z}^{(t)} \end{bmatrix}^\mathsf{T} + \delta\boldsymbol{x}^{(t)}, \qquad (24)$$

until convergence. After convergence, the complete fruit model can be reconstructed using marching cubes with the optimized $\boldsymbol{z}^*$ at 3D grid queries in the fruit's canonical coordinate system. The reconstruction can then be transformed into the world coordinate system using $\mathsf{T}_{ow}^* = \exp(\boldsymbol{\xi}_{ow}^*)$.

## IV. EXPERIMENTAL EVALUATION

The main focus of this work is a pipeline for multi-resolution mapping in orchard environments including fruits shape completion and pose estimation. We present our experiments to show the capabilities of our method. The results of our experiments also support the claims that our approach is able to jointly (i) predict the 3D shape of fruits even under substantial occlusions in real commercial greenhouse environments and (ii) estimate the pose of each fruit in the 3D map. (iii) Additionally, our multi-resolution map representation yield substantial improvement in the shape completion task over fixed-resolution maps.
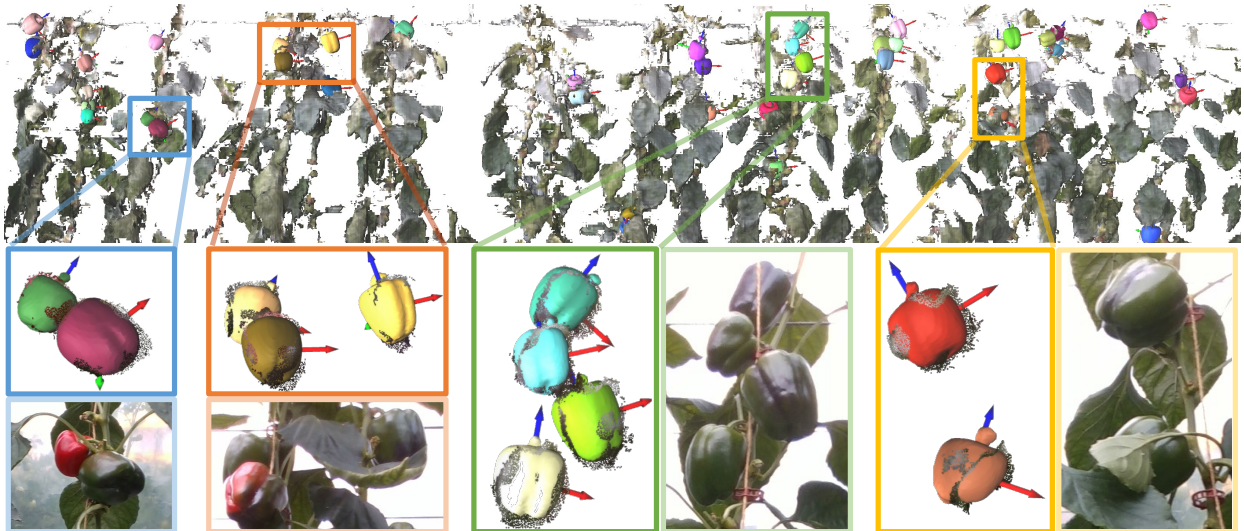
Fig. 7: Qualitative results of the panoptic mapping with fruit shape completion and pose estimation in a sweet pepper greenhouse. Our method successfully achieves convincing reconstructions and poses, even with partial observations and despite heavy occlusions. We also show a failure case in the orange box on the right, where noisy partial input leads to wrong pose of the sweet pepper on the top.

TABLE I: Reconstruction results in the commercial greenhouse. The ↓ and ↑ indicate that lower or higher values mean better performance.

| Approach | $D_C$ [mm] ↓ avg | f-score [%] ↑ avg | precision [%] ↑ avg | recall [%] ↑ avg | $E_{rot}$ [deg] ↓ avg | $E_{tran}$ [mm] ↓ avg | inference time [s] ↓ avg |
|---|---|---|---|---|---|---|---|
| CPD [20] | 25.38 | 3.09 | 8.10 | 1.92 | 26.79 | 27.74 | 0.57 |
| PF-SGD [17] | 9.28 | 35.03 | 37.32 | 33.21 | 29.61 | 19.73 | 30.21 |
| DeepSDF [24] | 9.33 | 35.24 | 32.38 | 38.77 | ✗ | ✗ | 16.01 |
| CoRe [15] | 6.90 | 41.47 | 43.17 | 41.64 | ✗ | ✗ | **0.004** |
| Ours | **5.29** | **58.56** | **61.28** | **56.26** | **11.48** | **11.20** | 0.62 |

## A. Experimental Setup

To showcase the capability of our proposed pipeline, we consider two typical and challenging crops, sweet pepper and strawberry, both of which have irregular shapes.

**Dataset:** For training and testing our shape completion network in a controlled environment we use the strawberries and sweet peppers dataset also used in our previous work [15]. We, additionally, collected a sweet pepper dataset with the robot shown in Fig. 1 in a greenhouse near Bonn, Germany, that we use for testing and evaluating our proposed solution in real conditions. This dataset contains RGB-D frames collected with the robot using an Intel RealSense d435i camera with a framerate of $15\,\mathrm{Hz}$. We then harvested about 50 sweet peppers present in the greenhouse and scanned them with a high-precision handheld LiDAR system as in Schunk et al. [29]. In this way, after manually aligning the fruit point cloud obtained with the two different sensors, we obtain the ground truth shape of each sweet pepper and also ground truth poses with respect to the fruit canonical pose, i.e., with the peduncle pointing upwards. See Fig. 6 for a visual impression of our greenhouse dataset.

**Metrics:** To evaluate our shape completion solution, we use the Chamfer distance $D_C$, i.e., the average symmetric squared distance of each point to its nearest neighbor in the other point cloud. We, additionally, use the f-score, precision, and recall at a fixed threshold ($\rho = 5\,\mathrm{mm}$ in our experiments) as proposed by Knapitsch et al. [10]. To evaluate the pose

estimated by our approach we report separate metrics for translations and rotations. We report the average translation error $E_{tran}$, i.e., the Euclidean distance between the predicted and the ground truth center, for each fruit. We define the rotation error $E_{rot}$ as the the intersection angle between the $z$-axis of predicted and ground truth pose. This metric ignores rotations around the fruit main axis as our target fruits are almost symmetric around it. Additionally, we report the average inference time needed to obtain the complete 3D shape with the estimated pose. In our experiments, we used an NVIDIA Quadro RTX A4000 GPU.

**Parameter settings:** We tune the hyperparameters of our system for better performance. We set the weight for each loss term in Eq. (10) as $w_s = 1.0$, $w_d = 0.05$, $w_m = 0.0002$, $w_r = 0.0005$ to ensure that the Hessian matrices of every loss term have the same order of magnitude. We set the damping factor $\lambda = 0.1$ in Eq. (11). To balance the efficiency and the performance, we sample $|\mathcal{P}_S| = 2,000$ points from each fruit submap. For each fruit instance on the image, we sample $G_f = 300$ and $G_b = 300$ pixels from the foreground and the background, respectively. For each ray, we sample $N = 30$ points on it. We set the occlusion and surface noise thresholds to $d_o = 3\,\mathrm{cm}$ and $\sigma = 1\,\mathrm{mm}$.

## B. Shape Completion and Pose Estimation

The first experiment evaluates the performance of our approach and supports the claims that our approach can

TABLE II: Reconstruction results in *controlled environment*. The ↓ and ↑ indicate that lower or higher values mean better performance.

| | Sweet Pepper | | | | | Strawberry | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Approach | $D_C$ [mm] ↓ avg | f-score [%] ↑ avg | precision [%] ↑ avg | recall [%] ↑ avg | time [s] ↓ avg | $D_C$ [mm] ↓ avg | f-score [%] ↑ avg | precision [%] ↑ avg | recall [%] ↑ avg | time [s] ↓ avg | learning | pose |
| CPD [20] | 12.36 | 39.84 | 76.68 | 27.07 | 15.62 | 5.13 | 57.93 | 94.09 | 42.34 | 0.57 | ✗ | ✓ |
| PF-SGD [17] | 3.97 | 68.95 | 71.20 | 66.94 | 17.48 | 2.71 | 86.08 | 88.82 | 83.90 | 8.10 | ✗ | ✓ |
| DeepSDF [24] | 29.78 | 37.12 | 32.96 | 46.06 | 44.13 | 3.61 | 74.01 | 83.76 | 68.32 | 36.84 | ✓ | ✗ |
| CoRe [15] | 7.83 | 52.85 | 47.38 | 60.00 | **0.004** | 2.67 | 86.01 | 87.97 | 84.85 | **0.004** | ✓ | ✗ |
| Ours | **3.16** | **80.86** | **82.14** | **79.72** | 0.60 | **2.42** | **92.81** | **94.38** | **91.53** | 0.53 | ✓ | ✓ |

TABLE III: Results of different maps in the greenhouse. The ↓ and ↑ indicate that lower or higher values mean better performance.

| Approach | $D_C$ [mm] ↓ avg | f-score [%] ↑ avg | precision [%] ↑ avg | recall [%] ↑ avg | $E_{rot}$ [deg] ↓ avg | $E_{tran}$ [mm] ↓ avg | inference time [s] ↓ avg | Online Map |
|---|---|---|---|---|---|---|---|---|
| Bundle Adjustment | **4.70** | **67.71** | **69.07** | **66.52** | **10.23** | **8.71** | 0.66 | ✗ |
| RGB-D Single Frame | 9.62 | 40.56 | 41.98 | 39.52 | 25.91 | 18.21 | **0.16** | ✓ |
| Fixed-Resolution Map | 7.29 | 44.22 | 46.29 | 42.62 | 17.48 | 16.02 | 0.75 | ✓ |
| Multi-Resolution Map (Ours) | 5.29 | 58.56 | 61.28 | 56.26 | 11.48 | 11.20 | 0.62 | ✓ |

predict the 3D shape of fruits under substantial occlusions and can estimate the pose of each fruit in the 3D map. The qualitative results are shown in Fig. 7. We report in Tab. I the metrics regarding shape completion and fruit pose estimation in real greenhouses. Our approach yields better performances in all metrics except for the inference time where we can still estimate poses and shapes of two fruit per second. The most competitive baselines for shape completion [15], [24] provide Chamfer distance performances of 1.6 mm and 4.0 mm worse than our approach and can not estimate fruit poses, thus limiting their applicability in this scenario. On the other hand, the best baseline that can estimate both shape and pose provide a Chamfer distance of about 4 mm, a rotation error of 10° and a translation error of about 8 mm worse than ours.

For a deeper comparison with the baselines and for testing our approach on a different fruit species, we evaluate the shape completion accuracy in a controlled environment in the laboratory where we estimate shapes of sweet peppers and strawberries, see Tab. II. Our approach yields better reconstruction performances on both species with an f-score of 80.86% on the sweet pepper and of 92.81% on the strawberry, while the closest baseline [17] reaches 68.95% and 86.08% respectively. In terms of execution time, our approach can estimate the 3D shape of roughly two fruit per second, note that the baseline [15] with lower inference time produce worse 3D shapes.

### C. Influence of Map Representation on Completion Results

The second experiment evaluates the effects of the map representation on estimating shapes and poses of fruits. This experiment illustrates that our multi-resolution map representation yield substantial improvement in the shape completion task over fixed-resolution maps. In Tab. III, we compare the shape completion results obtained with using our multi-resolution mapping strategy (1 cm / 3 mm voxel size) against the results of fixed-resolution maps (1 cm voxel size) and single RGB-D frames. Note that our multi-resolution mapping system can run in real-time. We additionally report the shape completion results obtained with a map obtained offline using bundle-adjustment software. For the Chamfer distance, our map yields a 28% improvement over fixed-resolution

TABLE IV: Ablation studies on the loss terms

| Loss | $D_C$ [mm] ↓ avg | f-score [%] ↑ avg | $E_{rot}$ [deg] ↓ avg | $E_{tran}$ [mm] ↓ avg |
|---|---|---|---|---|
| No $\mathcal{L}_s$ | 8.17 | 38.71 | 15.80 | 17.99 |
| No $\mathcal{L}_d$ | 8.93 | 36.12 | 17.11 | 19.78 |
| No $\mathcal{L}_m$ | 10.09 | 35.89 | 19.16 | 23.34 |
| No $\mathcal{L}_r$ | 7.64 | 40.24 | 15.42 | 12.34 |
| All Loss (Ours) | **5.29** | **58.56** | **11.48** | **11.20** |

map and a 45% improvement over single RGB-D frame. Considering the poses, using our map yields an improvement of 6° and 5 mm with respect to fixed-resolution map and an improvement of 14° and 7 mm for over single RGB-D frame. Notably, by using our online mapping strategy, we obtain competitive results with respect to using an offline map built by photogrammetric bundle adjustment. Our completion results are only 0.6 mm less accurate than the one obtained by using the offline method. Similarly, our predicted pose is around 1° and 3 mm less accurate.

### D. Ablation Study

We ran several ablation studies to evaluate the impact of each loss term in Eq. (10). From the metrics in Tab. IV, it is clear that our loss design with all four terms yields the best performances in both shape completion and pose estimation. As expected, the regularization term $\mathcal{L}_r$ has the lowest impact in the final results. Instead, the biggest impact on the performances is given by the mask term $\mathcal{L}_m$ defined in Eq. (9). Without this term, the f-score drops to 35.89% instead of 58.56%, the rotation error increases from 11.48° to 19.16° and the translation error from 11.20 mm to 23.34 mm.

### V. CONCLUSION

We presented a novel approach for panoptic mapping in real agricultural greenhouses using a mobile robot equipped with RGB-D cameras. We build a multi-resolution map to represent fruits with higher resolution and jointly estimate complete 3D shapes of fruits and their pose in the 3D map. Our method exploits high-precision 3D scanning to learn a general fruit shape prior that we use at inference time together with an occlusion-aware differentiable rendering pipeline. This allows us to successfully complete partial fruit observations and estimate the 7 DoF pose of each fruit in

the map. We solve the joint shape and pose optimization efficiently with analytical Jacobians, allowing for its application online. We evaluated our approach on different datasets and provided comparisons to other existing techniques and supported all our claims made. The experiments suggest that the proposed approach yields higher shape completion and pose estimation accuracy than existing baselines. It provides precise and complete fruits models, allowing farmers to assess the status of the orchard. Furthermore, our map providing fruits shape and pose explicitly can serve as a basis for planning autonomous fruit harvesting missions.

## REFERENCES

[1] A. Ahmadi, M. Halstead, and C. McCool. Bonnbot-i: A precise weed management and crop monitoring platform. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2022.

[2] B. Arad, J. Balendonck, R. Barth, O. Ben-Shahar, Y. Edan, T. Hellström, J. Hemming, P. Kurtser, O. Ringdahl, T. Tielen, et al. Development of a sweet pepper harvesting robot. *Journal of Field Robotics (JFR)*, 37(6):1027–1039, 2020.

[3] T. Botterill, S. Paulin, R. Green, S. Williams, J. Lin, V. Saxton, S. Mills, X. Chen, and S. Corbett-Davies. A robot system for pruning grape vines. *Journal of Field Robotics (JFR)*, 34(6):1100–1122, 2017.

[4] J. Gall, C. Stoll, E. De Aguiar, C. Theobalt, B. Rosenhahn, and H. 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.

[5] M. Halstead, A. Ahmadi, C. Smitt, O. Schmittmann, and C. McCool. Crop agnostic monitoring driven by deep learning. *Frontiers in plant science*, 12, 2021.

[6] M. Halstead, S. Denman, C. Fookes, and C. McCool. Fruit detection in the wild: The impact of varying conditions and cultivar. In *Proc. of Digital Image Comp.: Techniques and Applications (DICTA)*, 2020.

[7] M. Halstead, C. McCool, S. Denman, T. Perez, and C. Fookes. Fruit quantity and ripeness estimation using a robotic vision system. *IEEE Robotics and Automation Letters (RA-L)*, 3(4):2995–3002, 2018.

[8] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. In *Proc. of the IEEE Intl. Conf. on Computer Vision (ICCV)*, 2017.

[9] J. Kierdorf, I. Weber, A. Kicherer, L. Zabawa, L. Drees, and R. Roscher. Behind the leaves: Estimation of occluded grapevine berries with conditional generative adversarial networks. *Frontiers in Artificial Intelligence*, 5:830026, 2022.

[10] 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.

[11] C. Lehnert, A. English, C. McCool, A. Tow, and T. Perez. Autonomous Sweet Pepper Harvesting for Protected Cropping Systems. *IEEE Robotics and Automation Letters (RA-L)*, 2(2):872–879, 2017.

[12] C. Lehnert, I. Sa, C. McCool, B. Upcroft, and T. Perez. Sweet Pepper Pose Detection and Grasping for Automated Crop Harvesting. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2016.

[13] W. Lorensen and H. Cline. Marching Cubes: a High Resolution 3D Surface Construction Algorithm. In *Proc. of the Intl. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pages 163–169, 1987.

[14] F. Magistri, N. Chebrolu, J. Behley, and C. Stachniss. Towards In-Field Phenotyping Exploiting Differentiable Rendering with Self-Consistency Loss. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2021.

[15] F. Magistri, E. Marks, S. Nagulavancha, I. Vizzo, T. Läbe, J. Behley, M. Halstead, C. McCool, and C. Stachniss. Contrastive 3d shape completion and reconstruction for agricultural robots using rgb-d frames. *IEEE Robotics and Automation Letters (RA-L)*, 7(4):10120–10127, 2022.

[16] S. Marangoz, T. Zaenker, R. Menon, and M. Bennewitz. Fruit mapping with shape completion for autonomous crop monitoring. In *Proc. of the International Conf. on Automation Science and Engineering (CASE)*, 2022.

[17] E. Marks, F. Magistri, and C. Stachniss. Precise 3d reconstruction of plants from uav imagery combining bundle adjustment and template matching. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[18] C. McCool, J. Beattie, J. Firn, C. Lehnert, J. Kulk, R. Russell, T. Perez, and O. Bawden. Efficacy of mechanical weeding tools: A study into alternative weed management strategies enabled by robotics (2). *IEEE Robotics and Automation Letters (RA-L)*, 3(2):1184–1190, 2018.

[19] C. McCool, I. Sa, F. Dayoub, C. Lehnert, T. Perez, and B. Upcroft. Visual Detection of Occluded Crop: For Automated Harvesting. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2016.

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

[21] R.A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A.J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinect-Fusion: Real-Time Dense Surface Mapping and Tracking. In *Proc. of the Intl. Symposium on Mixed and Augmented Reality (ISMAR)*, 2011.

[22] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwar, and J. Nieto. Voxblox: Incremental 3d euclidean signed distance fields for on-board mav planning. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 1366–1373, 2017.

[23] Y. Pan, Y. Kompis, L. Bartolomei, R. Mascaro, C. Stachniss, and M. Chli. Voxfield: Non-projective signed distance fields for online planning and 3d reconstruction. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2022.

[24] J.J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[25] A. Rosinol, M. Abate, Y. Chang, and L. Carlone. Kimera: an open-source library for real-time metric-semantic localization and mapping. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[26] M. Runz, K. Li, M. Tang, L. Ma, C. Kong, T. Schmidt, I. Reid, L. Agapito, J. Straub, S. Lovegrove, et al. Frodo: From detections to 3d objects. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[27] I. Sa, C. Lehnert, A. English, C. McCool, F. Dayoub, B. Upcroft, and T. Perez. Peduncle Detection of Sweet Pepper for Autonomous Crop Harvesting - Combined Colour and 3D Information. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2017.

[28] L. Schmid, J. Delmerico, J. Schönberger, J. Nieto, M. Pollefeys, R. Siegwart, and C. Cadena. Panoptic multi-tsdfs: a flexible representation for online multi-resolution volumetric mapping and long-term dynamic scene consistency. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[29] D. Schunck, F. Magistri, R. Rosu, A. Cornelißen, N. Chebrolu, S. Paulus, J. Léon, S. Behnke, C. Stachniss, H. Kuhlmann, and L. Klingbeil. Pheno4D: A spatio-temporal dataset of maize and tomato plant point clouds for phenotyping and advanced plant analysis. *PLOS ONE*, 16(8):1–18, 2021.

[30] C. Smitt, M. Halstead, T. Zaenker, M. Bennewitz, and C. McCool. Pathobot: A robot for glasshouse crop phenotyping and intervention. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2021.

[31] A. Tagliasacchi, M. Schröder, A. Tkach, S. Bouaziz, M. Botsch, and M. Pauly. Robust articulated-icp for real-time hand tracking. In *Computer Graphics Forum*, volume 34, pages 101–114, 2015.

[32] N. Wagner, R. Kirk, M. Hanheide, G. Cielniak, et al. Efficient and robust orientation estimation of strawberries for fruit picking applications. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2021.

[33] J. Wang, M. Rünz, and L. Agapito. Dsp-slam: Object oriented slam with deep shape priors. In *Proc. of the Intl. Conf. on 3D Vision (3DV)*, 2021.

[34] J. Yi, L. Krusenbaum, P. Unger, H. Hüging, S.J. Seidel, G. Schaaf, and J. Gall. Deep learning for non-invasive diagnosis of nutrient deficiencies in sugar beet using rgb images. *Sensors*, 20(20):5893, 2020.

[35] W. Yuan, T. Khot, D. Held, C. Mertz, and M. Hebert. Pcn: Point completion network. In *Proc. of the Intl. Conf. on 3D Vision (3DV)*, 2018.

[36] D. Zermas, V. Morellas, D. Mulla, and N. Papanikolopoulos. 3d model processing for high throughput phenotype extraction–the case of corn. *Computers and Electronics in Agriculture*, 172:105047, 2020.