

# Fruit Mapping with Shape Completion for Autonomous Crop Monitoring

Salih Marangoz

Tobias Zaenker

Rohit Menon

Maren Bennewitz

**Abstract**— Autonomous crop monitoring is a difficult task due to the complex structure of plants. Occlusions from leaves can make it impossible to obtain complete views about all fruits of, e.g., pepper plants. Therefore, accurately estimating the shape and volume of fruits from partial information is crucial to enable further advanced automation tasks such as yield estimation and automated fruit picking. In this paper, we present an approach for mapping fruits on plants and estimating their shape by matching superellipsoids. Our system segments fruits in images and uses their masks to generate point clouds of the fruits. To combine sequences of acquired point clouds, we utilize a real-time 3D mapping framework and build up a fruit map based on truncated signed distance fields. We cluster fruits from this map and use optimized superellipsoids for matching to obtain accurate shape estimates. In our experiments, we show in various simulated scenarios with a robotic arm equipped with an RGB-D camera that our approach can accurately estimate fruit volumes. Additionally, we provide qualitative results of estimated fruit shapes from data recorded in a commercial glasshouse environment.

## I. INTRODUCTION

In the context of increasing demand for seasonal migrant farm labour in member European Union states and its supply side disruption due to the pandemic [1], autonomous crop monitoring has gained further importance for plant phenotyping, yield monitoring, crop protection, and harvesting. The availability of different sensors and systems has enabled crop monitoring at different levels from global low resolution satellite mapping [2] to drone-based crop-level mapping [3], [4] to individualized plant level mapping using mobile robots [5] and mobile manipulators [6], with an increase in the level of detail and precision. While global crop monitoring is necessary to make coarse yield predictions, fruit mapping is key to accurate yield monitoring and to performing automated precise interventions.

Fruit mapping consists of the localization of fruits and estimation of their shape. With the advent of inexpensive RGB-D cameras, fusing instance masks from panoptic segmentation networks with depth information provides 3D information about the size and location of the fruits. However, autonomous mapping of crops such as sweet peppers can be a challenging task since the plants have complex structures, with many leaves that may partially occlude fruits. Therefore, obtaining accurate information about the location and shape of fruits can be difficult. Even if observations from multiple views are acquired, fruits can often be only partially covered.

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany’s Excellence Strategy – EXC 2070 – 390732324. All authors are with the Humanoid Robots Lab, University of Bonn, Germany.

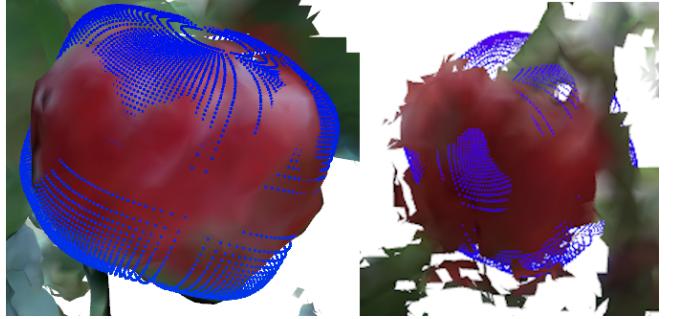


Fig. 1: Illustration of matched superellipsoids to fruits. Map generated from multiple views in a commercial glasshouse environment. Estimated superellipsoid surfaces are visualized as blue point clouds. While the left fruit has been observed from almost all directions, only half of the right fruit has been mapped. In both cases, our system was able to accurately predict their shape.

To enable shape prediction in these cases, we present a system to estimate the shape of both partially and fully observed fruits. We build upon ideas of Lehnert *et al.* [7] who proposed to match the shape of sweet peppers with superellipsoids. Whereas their experiments were limited to controlled environments with single pepper plants, we integrated our superellipsoid matching system with our previously developed viewpoint planner [8]. As a result, we are able to apply our fruit mapping system on data captured with an RGB-D camera on a robotic arm in various scenarios containing multiple plants with sweet peppers.

Fig. 1 shows an example application of our system using data of sweet pepper crops in a glasshouse. As can be seen, fruit masks are detected in the RGB image to subsequently generate a fruit point cloud from depth data. From a sequence of point clouds, we generate a fruit map in form of truncated signed distance fields (TSDF) with the mapping framework voxblox [9]. The surface cloud of this TSDF map is clustered to separate fruits, and superellipsoids are fitted by first estimating center and normals of the points and then minimizing the distance of detected points to the superellipsoid surface.

Our main contributions are the following:

- A complete framework that accurately maps and estimates fruit shapes using multiple views,
- An implementation in ROS, providing easy integration with our previously developed ROI viewpoint planner,
- An evaluation of the accuracy of the estimated fruit volumes in various simulated scenarios with multiple crops,
- Qualitative results on data recorded in a commercial glasshouse environment.

## II. RELATED WORK

Several approaches exist that are based on reconstruction using 3D data collected from different viewpoints to obtain a full 3D model, however, with no explicit modeling of shape. For example, Wang *et al.* [10] developed automatic fruit shape measurement based on 3D reconstruction, using a Kinect sensor and an electric turntable. Similarly, Jadhav *et al.* [11] investigated the volumetric 3D reconstruction of different fruits based on voxel mapping using silhouette probabilities and inter-image homographies, with multiple RGB cameras and a checkerboard. Complete reconstruction approaches are generally suitable for ground truth dataset generation or volumetric estimation in laboratory conditions. However, the reconstruction methods are computationally expensive and are generally not suitable for resource-constrained, real-time agriculture automation tasks. Moreover, occlusions due to the plant leaves or unreachability of the sensor to cover the complete view can lead to incomplete scanning of the fruits.

Viewpoint planning uses the hitherto accumulated information to plan the next best view for maximising the information gain. As an example, Roy and Isler [12] used a viewpoint planning approach to count apples in orchards, however, without any volumetric estimation. In our previous work [8], we formulated a sampling approach based on regions of interest (ROI), i.e., fruits that outperforms those that do not consider ROIs, with respect to the number of detected fruits and the estimated volume. However, the previous approach suffered from inaccuracies in fruit volume estimation as no shape estimation or modeling was carried out for predicting the unobserved fruit regions.

State-of-the-art shape completion or fruit modeling approaches do not use a viewpoint planner for maximizing the information gain, instead they typically attempt to predict fruit shape and pose using data from a single view or few views. Kang *et al.* [13] used the DasNet deep learning network [14] for fruit recognition, and 3D Sphere Hough Transform to estimate the fruit pose and size, for autonomous apple harvesting. Similarly, Lin *et al.* [15] demonstrated that sphere fitting works better than a bounding box method for pose estimation of guavas. However, while spherical fitting provides acceptable results for autonomous harvesting, the reduction of different fruits to simple spheres is neither generalizable, nor reliable for localized yield predictions. Ge *et al.* [16] formulated another approach based on shape from symmetry [17], to generate and rate different hypotheses for the symmetrical plane and the mirrored point sets, for the partial view, to reconstruct the complete shape of strawberry fruits for fruit localization. Furthermore, photogrammetry-based methods have been used for estimating real length and width after fitting ellipses for apples and mangoes, respectively, in the 2D plane [18], [19].

The most closely related work is the approach by Lehnert *et al.* [7] that fits superellipsoids to sweet peppers. Here, point clouds from different viewpoints were merged into a single point cloud using different registration methods of

which Kinect Fusion (KinFu) was found to perform best at pose tracking. After registration and segmentation of red sweet papers based on color using the HSV colour space, sweet pepper pose estimation is performed using constrained non-linear least squares optimization. A superellipsoid model is then used for estimating parameters of the fruit shape and for obtaining the 6 DoF transform between the registered point cloud and the model.

KinFu can be used for realtime mapping of scenes while concurrently obtaining the camera pose by tracking the depth frames continuously as in [7]. However, as the viewpoint planner used in our previous approach [8] plans discrete viewpoints for maximising information gain, KinFu cannot perform continuous tracking and registration which needs small movements in successive frames for the convergence of ICP alignment [20]. Oleynikova *et al.* [9] developed voxblox to build TSDFs based map representations for local planning of Micro Aerial Vehicles. Unlike KinFu, which requires a known map size, large amounts of memory, and GPU computational power, voxblox can dynamically grow maps as it adds point cloud data from different viewpoints. As voxblox can run on a single CPU core with low memory requirements and still build TSDF maps faster than building occupancy maps, we replaced the 3D occupancy maps in [21] with TSDF maps.

In conclusion, most state-of-the-art approaches for fruit mapping fall into three categories – naive complete reconstruction methods, viewpoint planning without any shape completion, and geometric shape completion methods. Our approach combines a viewpoint planner with shape completion for mapping of fruits for estimation of count and volume, in the wild, with an improved TSDF based mapping.

## III. SYSTEM OVERVIEW

Our approach aims at estimating the shape of detected fruits, which might be partially occluded. We use our previously developed viewpoint planner [8] to move a camera placed on a robotic arm, which enables us to observe fruits from different perspectives. In order to estimate the shape of detected fruits, we build a map of the environment. Previously, we generated an octree with marked regions of interest (ROIs), i.e., fruits, to guide the planner and estimate the fruit position and size. While this enabled ROI targeted viewpoint planning and a rough estimation of fruit positions, due to the limited accuracy of the octree map, the shape of fruits could not be accurately determined.

For the new approach presented in this paper, we therefore build a separate map of the environment specifically for the fruits. We use the masks of detected fruits to generate a separate fruit point cloud as input for the fruit mapping. This point cloud is processed to remove outliers and forwarded to the voxblox mapping system [9], which creates a TSDF map of the fruits. Our superellipsoid matching system receives the surface point cloud generated from that map, clusters it into individual fruits, and optimizes the parameters of a superellipsoid to find the best fitting shape for the corresponding points. The parameters of the superellipsoid can be used to

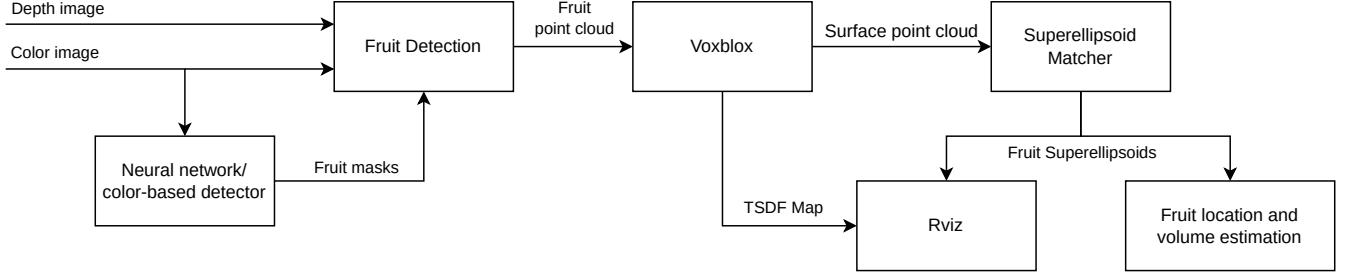


Fig. 2: Overview of our system. Color and depth images are converted into a fruit point cloud from generated fruit masks, which is processed by voxblox [9] into a truncated signed distance field (TSDF) map. The surface point cloud from this map is used to fit superellipsoids to individual fruits, which can be visualized together with the map, or used to estimate position and volume.

compute the fruit position and volume. Additionally, a fruit point cloud for visualization purposes is generated. Fig. 2 shows an overview of the approach described above.

#### IV. FRUIT MAPPING AND SHAPE ESTIMATION

##### A. Fruit Detection and Mapping

In order to match superellipsoid to fruits, they have to be detected first. Like in our previous work, we provide two methods to detect fruits in images. The first one is a color threshold for detecting red pixels, and is used in our simulated experiments with only red peppers. For our real world data, we utilize a neural network trained on a dataset recorded at the University of Bonn [22] to detect the fruit masks. In both cases, we end with a set of pixels belonging to fruits. We match those with the point cloud generated from the depth image of the RGB-D camera, and extract a cloud only containing the fruit points. This cloud is used to generate the map used for our shape completion, while the complete cloud can be used to generate an additional map containing full plants. Before that, outliers are removed using statistical outlier removal to counter depth noise.

For generating the map used to match superellipsoids to fruits, we utilize the mapping framework voxblox [9]. We configure voxblox with a TSDF voxel size of 0.004 m and 64 voxels per side. We disable ICP tracking feature of voxblox and utilize position feedbacks from the robot arm.

##### B. Clustering and Shape Estimation

From the fruit TSDF map generated with voxblox, we extract the surface point cloud to match our fruit predictions. In a first processing step, we extract clusters to identify individual fruits. We utilize Euclidean cluster extraction with a cluster tolerance of 0.01 m, a minimum cluster size of 100 points and a maximum cluster size of 10000 points for that. For each extracted cluster, we first estimate the normals of the points. Taking into account the normals, a cluster center  $w$  is estimated using a closed form least-squares solution for minimizing the sum of perpendicular distances from the estimated point to all the lines [23]. We also use L2 regularization to bias the solution towards a reference point (i.e., the mean of all points in a cluster) with the regularization value of  $\lambda = 2.5$  [23]. This center is used as start point for the optimization of the superellipsoid.

As in [7], we represent superellipsoids with the following equation:

$$f(x, y, z) = \left[ \left( \frac{x}{a} \right)^{\frac{2}{\varepsilon_2}} + \left( \frac{y}{b} \right)^{\frac{2}{\varepsilon_2}} \right]^{\frac{\varepsilon_2}{\varepsilon_1}} + \left( \frac{z}{c} \right)^{\frac{2}{\varepsilon_1}} = 1 \quad (1)$$

The parameters  $a$ ,  $b$ ,  $c$ ,  $\varepsilon_1$ , and  $\varepsilon_2$ , together with the translation  $t_x$ ,  $t_y$ ,  $t_z$  and rotation  $\phi$ ,  $\theta$ ,  $\psi$  make up 11 parameters to optimize. We set the initial value of  $t_x$ ,  $t_y$ ,  $t_z$  to the estimated center and the rotation to 0. For the superellipsoid parameters, we start with initial values of  $a = b = c = 0.05$  m and  $\varepsilon_1 = \varepsilon_2 = 0.5$ .

Compared to [7], we changed the cost function to achieve better fits. Jaklič *et al.* [24] describe approximated distance of a point to a superellipsoid as:

$$d = |r_0 - \beta r_0| = |r_0| \cdot |1 - f^{-\frac{\varepsilon_1}{2}}(x_0, y_0, z_0)| \quad (2)$$

where  $r_0 = (x_0, y_0, z_0)$  represents the coordinates of a point with respect to the superellipsoid center.  $\beta$  is the scaling factor to project  $r_0$  to the superellipsoid surface, and computed by solving  $f(\beta x_0, \beta y_0, \beta z_0)$  for  $\beta$ . To find the best fitting shape, we minimize the sum of  $d$  over all points. We get the following optimization task, with priors encouraging the superellipsoid center to be closer to the estimated center and the  $a$ ,  $b$ , and  $c$  parameters to be close to each other:

$$\min_{a, b, c, t_x, t_y, t_z, \phi, \theta, \psi} \sum_i \left[ d_i^2 + \alpha \|t - w\|_2^2 + \gamma ((a-b)^2 + (b-c)^2 + (c-a)^2) \right] \quad (3)$$

We apply the Levenberg–Marquardt algorithm to optimize the parameters. The numerical computations are performed using the open source Ceres solver [25].

We limit  $a$ ,  $b$ , and  $c$  between 0.02 and 0.15 m, and  $\varepsilon_1$  and  $\varepsilon_2$  between 0.3 and 0.9, to avoid unrealistically small, large or deformed superellipsoids that don't realistically represent a fruit. We set regularization constants  $\alpha = \gamma = 0.1$ . We also limit the maximum number of iterations to 100 and discard optimizations that terminated without convergence.

##### C. Shape Evaluation

Our node provides the results of the matched superellipsoids in multiple forms. For visualization, surface point

clouds are generated from the optimized parameters. We provide a colored point cloud with different colors for each cluster, as well a labeled point cloud including the computed normals. For further usage, the superellipsoid parameters and volume are published. We compute the volume using the following formula, as described in [24]:

$$V = 2abc\varepsilon_1\varepsilon_2 B\left(\frac{\varepsilon_1}{2} + 1, \varepsilon_1\right) B\left(\frac{\varepsilon_2}{2}, \frac{\varepsilon_2}{2}\right); \quad (4)$$

where  $B$  is the beta function:

$$B(x, y) = 2 \int_0^{\frac{\pi}{2}} \sin^{2x-1} \phi \cos^{2y-1} \phi d\phi = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)} \quad (5)$$

For our simulated environments, we extract a point cloud from the mesh models of the fruits and optimize superellipsoid parameters to fit this point cloud for each fruit. We read the position and orientation of the plants directly from the Gazebo simulator and can therefore generate a ground truth superellipsoid for each fruit in the scene.

We can match ground truth and detected ellipsoids by comparing the distance of the centers and the volumes.

## V. EXPERIMENTS

We carried out experiments with of our shape estimation method in a simulated environment for quantitative analysis and with data collected from the glasshouse to validate our approach in real world conditions.

For our evaluation in simulated scenarios, we used the Gazebo based simulation setup presented in [8], but with an additional new environment with more and different plants. It simulates a UR5e arm from Universal Robots with an RGB-D camera placed at its flange. We configured the planner to move the arm to new views autonomously.

We placed the arm on a static pole for the first scenario, and for the other two, it was attached to a retractable pole hanging from the ceiling, to enable good coverage of the fruits. The three scenarios are described with pictures in Figures 3a to 3c.

### A. Simulated Scenarios

To evaluate the accuracy of the fruit shape estimation with our new method to our old approach, we compare the average estimated fruit center distance and volume accuracy of the detected fruits. Additionally, we set a new restriction on the detected fruits – they are counted only when the specified minimum accuracy criterion is met. For the evaluation, we consider the number of fruits with minimum accuracy 0 (i.e. the detected volume must not be more than twice the ground truth volume) and minimum accuracy 0.5 (the detected volume must not be more than 50% larger or smaller than the ground truth.)

This gives us the following metrics:

- *Detected fruits*: Number of found fruits that can be matched with ground truth fruits, which means that their center distance is less than 20 cm, and their volume accuracy is greater than the specified minimum.

- *Center distance*: Average distance of the fruit centers from the ground truth centers.
- *Volume accuracy*: Average accuracy of the cluster volumes. The accuracy is computed as follows:

$$acc_V = 1 - \frac{|V_{det} - V_{gt}|}{V_{gt}} \quad (6)$$

where  $V_{det}$  is the detected volume and  $V_{gt}$ , the ground truth volume. For detected volumes less than the ground truth volume, the  $acc_V$  varies from 0 to 1, with 1 representing the best possible accuracy when they are equal. On the other hand, for  $V_{det}$  greater than  $2V_{gt}$ , the  $acc_V$  turns negative. For our old method, the volumes are computed as bounding box volumes of the detected clusters. For the new method, we use the superellipsoid volume as described in Section IV-C.

For each scenario, we executed 20 trials, each with a planning time of three minutes.

Figures 3d to 3f illustrate the number of detected fruits over the three minutes planning time, and Figures 3g to 3i show the quantitative results for the three scenarios.

While the number of detected fruits with no volume accuracy bounds is similar to our old approach, our current approach with the superellipsoid matcher recognizes significantly greater number of fruits, when the minimum volume accuracy bound is applied. With a minimum accuracy of 0.5, we detect an average of 7.7 fruits with our new approach compared to 4.7 with the old approach in Scenario 1, 17.7 compared to 6.0 in Scenario 2, and 27.2 compared to 12.4 in Scenario 3.

It can also be seen that with our old approach, the number of detected fruits with minimum accuracy drops towards the end. This can happen if nearby fruits are combined to a single fruit, which greatly decreases the volume accuracy. Since our new method uses a more accurate map and better clustering, this happens less often, leading to a considerable improvement in count and volume estimation.

The average center distances are similar for both approaches. However, the average volume accuracy of our superellipsoid matcher is much higher. When counting all detected fruits, with average accuracies between 0.22 and 0.35, it is still quite low. This is because the clustering still can fail if there are not enough fruit points detected and nearby fruits are merged, leading to overestimated shapes with negative accuracies, which bring down the average. When removing those outliers, which was done via prior assumptions about the maximum fruit size, the average accuracy increases appreciably, with accuracies between 0.67 and 0.70 compared to only between 0.35 and 0.45 with our old approach. This accuracy covers on average over 70% of the detected fruits in Scenario 1 and over 80% in Scenario 2 and 3.

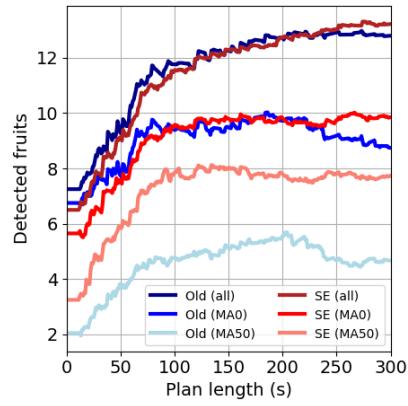
The evaluation demonstrates that our approach with superellipsoid based shape completion can lead to a higher number of accurately detected fruits. While there are still some outliers, the volume accuracy for most of the sweet peppers shows that they can be satisfactorily approximated



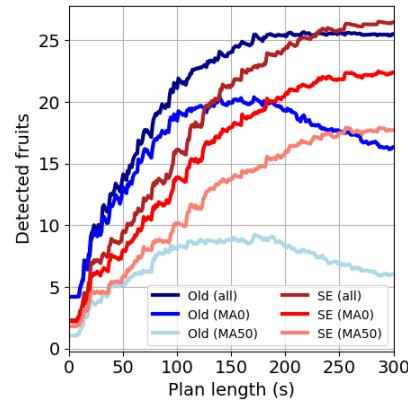
(a) Scenario 1

(b) Scenario 2

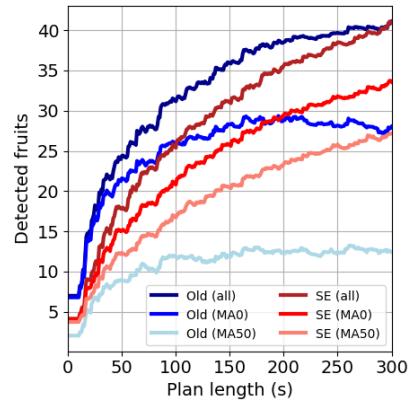
(c) Scenario 3



(d) Fruits Scenario 1



(e) Fruits Scenario 2



(f) Fruits Scenario 3

	Fruits	Center	Vol.
Old (all)	12.8 $\pm 0.9$	2.5 $\pm 0.5$	-0.12 $\pm 0.79$
SE (all)	13.2 $\pm 0.9$	2.9 $\pm 0.7$	0.26 $\pm 0.27$
Old (MA0)	8.7 $\pm 2.6$	2.6 $\pm 1.1$	0.45 $\pm 0.12$
SE (MA0)	9.8 $\pm 1.9$	2.3 $\pm 0.8$	0.69 $\pm 0.06$
Old (MA50)	4.7 $\pm 2.2$	1.6 $\pm 0.8$	0.65 $\pm 0.17$
SE (MA50)	7.7 $\pm 1.9$	1.9 $\pm 0.9$	0.81 $\pm 0.04$

(g) Results Scenario 1

	Fruits	Center	Vol.
Old (all)	25.5 $\pm 1.0$	2.2 $\pm 0.3$	-0.05 $\pm 0.21$
SE (all)	26.5 $\pm 1.2$	1.8 $\pm 0.6$	0.35 $\pm 0.23$
Old (MA0)	16.4 $\pm 3.4$	2.1 $\pm 0.5$	0.35 $\pm 0.06$
SE (MA0)	22.4 $\pm 2.4$	1.4 $\pm 0.6$	0.67 $\pm 0.04$
Old (MA50)	6.0 $\pm 2.0$	1.8 $\pm 0.4$	0.66 $\pm 0.07$
SE (MA50)	17.7 $\pm 2.1$	1.2 $\pm 0.4$	0.78 $\pm 0.03$

(h) Results Scenario 2

	Fruits	Center	Vol.
Old (all)	41.1 $\pm 3.3$	2.4 $\pm 0.4$	-0.19 $\pm 0.27$
SE (all)	40.9 $\pm 3.9$	1.9 $\pm 0.4$	0.22 $\pm 0.22$
Old (MA0)	28.0 $\pm 4.7$	2.5 $\pm 0.5$	0.44 $\pm 0.06$
SE (MA0)	33.6 $\pm 3.9$	1.3 $\pm 0.3$	0.70 $\pm 0.04$
Old (MA50)	12.4 $\pm 3.6$	2.1 $\pm 0.6$	0.71 $\pm 0.04$
SE (MA50)	27.2 $\pm 3.9$	1.2 $\pm 0.3$	0.80 $\pm 0.02$

(i) Results Scenario 3

Fig. 3: (a-c): Our three simulated scenarios. *Scenario 1*: Simulated environment with static arm, as used in [8], with 4 plants and 14 fruits. *Scenario 2*: Simulated environment with retractable arm, as in [8], with 4 plants and 28 fruits. *Scenario 3*: New simulated environment. It also uses the retractable arm, but contains a total of 12 plants, ordered in two rows, with different growth stages and fruits. There are a total of 47 red fruits in this environment.

(d-f): Results for the three simulated scenarios. For each tested approach, 20 trials with a duration of three minutes each were performed. The plots show the average number of detected fruits of our superellipsoid matcher (SE) compared to our old octree-based approach (**Old**), with no minimum accuracy (**all**), a minimum accuracy of 0 (**MA0**), and a minimum accuracy of 0.5 (**MA50**). As can be seen, while the number of detected fruits with no accuracy bounds is similar, our new approach detects much more fruits with good accuracy. (g-i): The average numerical results at the end of the episodes regarding number of detected fruits (**Fruits**), center distance in cm (**Center**), and volume accuracy (**Vol.**).



Fig. 4: Generated map with superellipsoids. Blue point clouds represent the estimated surface of the matched shapes.

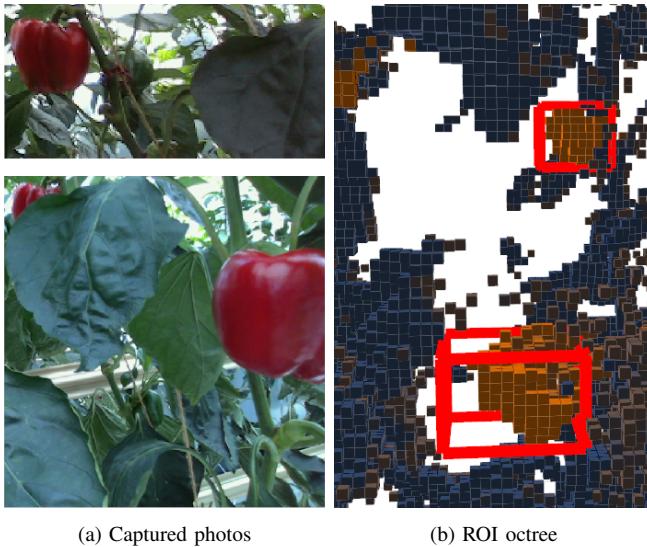


Fig. 5: Left: Captured RGB images of the glasshouse in the segment mapped in Figure 4 . Right: The octree-based map generated in [8], with red bounding boxes visualizing the matched fruit volume. The two fruits are the same as the two red fruits on the right in Figure 4

through superellipsoids, even if only partially observed by our viewpoint planner.

### B. Real-World Glasshouse Experiment

To evaluate how our approach handles real-word data, we utilized a recording of the data feed gathered from our previously developed viewpoint planner in a capsicum glasshouse. Color and depth images were generated using a RealSense L515 Lidar sensor, and the data includes the arm configuration and the camera poses.

The recorded images are passed through a neural network [22] to generate fruit masks which are fused with the depth images to generate the fruit point clouds used for voxblox mapping. In addition to the fruit map used to estimate shapes with superellipsoids, we generate a map from the full point cloud for visualization purposes.

In Figure 4, you can see the generated full TSDF map of a section of the glasshouse, with the superellipsoids matched from the fruit-only map visualized as blue point clouds. For comparison, Figure 5 shows the results of the same segment using the octree-based map presented in [8]. It can be seen

that our new method provides both a more accurate map and a better approximation of the fruit shapes.

### VI. CONCLUSIONS

To move a step closer to autonomous crop monitoring and yield estimation, we developed a framework to accurately estimate the shape of fruits such as sweet peppers. Our system is embedded into a viewpoint planning approach and builds up a map of the fruits in form of truncated signed distance fields, on which fruits are clustered. Subsequently, the shape of clustered fruits is approximated by fitting superellipsoids to the surface points. As the evaluation with a simulated robotic arm equipped with an RGB-D camera demonstrates, our system can detect a majority of the fruits with good accuracy, and the qualitative results on real-world data show that it is applicable under realistic conditions in commercial glasshouse environments.

In the future, we plan to use the estimated shapes to determine the missing parts of partially detected fruits to guide our viewpoint planner to achieve better coverage.

### REFERENCES

- [1] M.-L. Augère-Granier, “Migrant seasonal workers in the european agricultural sector,” *Briefing. EU: European Parliamentary Research Service*. [https://www.europarl.europa.eu/RegData/etudes/BRIE/2021/689347/EPRS\\_BRI\\_2021\\_689347\\_EN.pdf](https://www.europarl.europa.eu/RegData/etudes/BRIE/2021/689347/EPRS_BRI_2021_689347_EN.pdf), 2021.
- [2] F. Rembold, C. Atzberger, I. Savin, and O. Rojas, “Using low resolution satellite imagery for yield prediction and yield anomaly detection,” *Remote Sensing*, vol. 5, no. 4, pp. 1704–1733, 2013.
- [3] M. Maimaitijiang, V. Sagan, P. Sidike, A. M. Daloye, H. Erkbol, and F. B. Fritschi, “Crop monitoring using satellite/uav data fusion and machine learning,” *Remote Sensing*, vol. 12, no. 9, p. 1357, 2020.
- [4] N. Stefas, H. Bayram, and V. Isler, “Vision-based monitoring of orchards with uavs,” *Computers and Electronics in Agriculture*, vol. 163, p. 104814, 2019.
- [5] J. Qiao, A. Sasao, S. Shibusawa, N. Kondo, and E. Morimoto, “Mapping yield and quality using the mobile fruit grading robot,” *Biosystems Engineering*, vol. 90, no. 2, pp. 135–142, 2005.
- [6] A. Shafiekhan, S. Kadam, F. B. Fritschi, and G. N. DeSouza, “Vinobot and vinoculer: two robotic platforms for high-throughput field phenotyping,” *Sensors*, vol. 17, no. 1, p. 214, 2017.
- [7] 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)*. IEEE, 2016, pp. 2428–2434.
- [8] T. Zaenker, C. Smitt, C. McCool, and M. Bennewitz, “Viewpoint planning for fruit size and position estimation,” in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2021.
- [9] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, 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)*, 2017.
- [10] Y. Wang and Y. Chen, “Fruit morphological measurement based on three-dimensional reconstruction,” *Agronomy*, vol. 10, no. 4, p. 455, 2020.
- [11] T. Jadhav, K. Singh, and A. Abhyankar, “Volumetric estimation using 3d reconstruction method for grading of fruits,” *Multimedia Tools and Applications*, vol. 78, no. 2, pp. 1613–1634, 2019.
- [12] P. Roy and V. Isler, “Active view planning for counting apples in orchards,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 6027–6032.
- [13] H. Kang, H. Zhou, and C. Chen, “Visual perception and modeling for autonomous apple harvesting,” *IEEE Access*, vol. 8, pp. 62 151–62 163, 2020.
- [14] H. Kang and C. Chen, “Fruit detection, segmentation and 3d visualisation of environments in apple orchards,” *Computers and Electronics in Agriculture*, vol. 171, p. 105302, 2020.

- [15] G. Lin, Y. Tang, X. Zou, J. Xiong, and J. Li, “Guava detection and pose estimation using a low-cost rgb-d sensor in the field,” *Sensors*, vol. 19, no. 2, p. 428, 2019.
- [16] Y. Ge, Y. Xiong, and P. J. From, “Symmetry-based 3d shape completion for fruit localisation for harvesting robots,” *biosystems engineering*, vol. 197, pp. 188–202, 2020.
- [17] S. Thrun and B. Wegbreit, “Shape from symmetry,” in *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, vol. 2. IEEE, 2005, pp. 1824–1831.
- [18] A. Gongal, M. Karkee, and S. Amatya, “Apple fruit size estimation using a 3d machine vision system,” *Information Processing in Agriculture*, vol. 5, no. 4, pp. 498–503, 2018.
- [19] Z. Wang, K. B. Walsh, and B. Verma, “On-tree mango fruit size estimation using rgb-d images,” *Sensors*, vol. 17, no. 12, p. 2738, 2017.
- [20] M. Pirovano, “Kinfu—an open source implementation of kinect fusion+ case study: implementing a 3d scanner with pcl,” *UniMi, Tech. Rep.*, 2012.
- [21] T. Zaenker, C. Lehnert, C. McCool, and M. Bennewitz, “Combining local and global viewpoint planning for fruit coverage,” in *Proc. of the Europ. Conf. on Mobile Robotics (ECMR)*, 2021.
- [22] M. Halstead, S. Denman, F. Clinton, and C. McCool, “Fruit detection in the wild: The impact of varying conditions and cultivar,” in *Digital Image Computing: Techniques and Applications (DICTA)*, 2020.
- [23] J. Traa, “Least-squares intersection of lines,” *University of Illinois Urbana-Champaign (UIUC)*, 2013.
- [24] A. Jaklič, A. Leonardis, and F. Solina, *Superquadrics and Their Geometric Properties*. Dordrecht: Springer Netherlands, 2000, pp. 13–39. [Online]. Available: [https://doi.org/10.1007/978-94-015-9456-1\\_2](https://doi.org/10.1007/978-94-015-9456-1_2)
- [25] S. Agarwal, K. Mierle, and Others, “Ceres solver,” <http://ceres-solver.org>.