Self-Supervised Learning for Panoptic Segmentation of Multiple Fruit Flower Species

Abubakar Siddique, *Student Member, IEEE*, Amy Tabb, *Senior Member, IEEE*, and Henry Medeiros, *Senior Member, IEEE*

***Abstract*—Convolutional neural networks trained using manually generated labels are commonly used for semantic or instance segmentation. In precision agriculture, automated flower detection methods use supervised models and post-processing techniques that may not perform consistently as the appearance of the flowers and the data acquisition conditions vary. We propose a self-supervised learning strategy to enhance the sensitivity of segmentation models to different flower species using automatically generated pseudo-labels. We employ a data augmentation and refinement approach to improve the accuracy of the model predictions. The augmented semantic predictions are then converted to panoptic pseudo-labels to iteratively train a multi-task model. The self-supervised model predictions can be refined with existing post-processing approaches to further improve their accuracy. An evaluation on a multi-species fruit tree flower dataset demonstrates that our method outperforms state-of-the-art models without computationally expensive post-processing steps, providing a new baseline for flower detection applications.**

***Index Terms*—Agricultural automation, incremental learning, object detection, semantic scene understanding, segmentation and categorization.**

I. Introduction

COMPUTER vision algorithms are becoming increasingly popular in agricultural applications. Detecting and counting flowers is an important crop management activity to optimize fruit production [1]. Automatic bloom intensity estimation methods have the potential to reduce workloads in large production fields. Many machine vision approaches have been proposed to address the challenges of estimating crop yield. Most recent flower detection and counting methods based on deep learning models require a large amount of manually labeled training data to achieve acceptable performance [2], [3], [4]. Although weakly supervised approaches [5] can simplify the training of convolutional neural networks (CNNs), they are not particularly effective to adapt large-scale, pre-trained models to unseen object categories.

Data augmentation [6], [7] is a de facto standard technique to reduce the dependence on manual annotations when training deep neural networks. But in agricultural visual data, the appearance of objects of interest and the scene conditions vary significantly from one field to another. Besides, since agricultural production environments usually require images to be acquired from moving vehicles [2], [4], [8], the sun conditions and dense background clutter make this task challenging in terms of model generalization. Hence, we still need to generate enough manual labels for various species of crops to generalize the prediction models across species with significantly different appearance and backgrounds potentially comprised of semantically distinct elements.

Although deep CNNs can perform reasonably accurate pixel-level semantic predictions [2], [9], false alarms due to similarities between flowers, fruits at different stages of maturation, and background objects limit potential opportunities for the application of computer vision algorithms to agricultural automation tasks. Instance [10] and panoptic [11] segmentation models might be able to better identify individual flowers or clusters of flowers and thus improve detection performance.

To address the above challenges, inspired by the works presented in [2], [11], [12], we propose a novel self-supervised panoptic segmentation approach that leverages a small number of annotations for supervised learning (SL) and then adjusts the model to challenging unlabeled datasets. In summary, the main contributions of this work are:

- A robust self-supervised flower segmentation method that addresses typical agricultural visual data challenges in fruit tree orchards.
- A novel panoptic pseudo-label generation technique for automatically updating the model for unlabeled datasets that contain severe clutter and illumination challenges.
- A robust sliding-window-based training and testing approach that does not require additional post processing to refine the network predictions.
- Extensive evaluations on multiple-species datasets, which demonstrate superior generalized performance over state-of-the-art techniques.
- Our source code and pre-trained models are available at https://github.com/siddiquemu/ssl_flower_semantic.

Manuscript received 7 June 2022; accepted 29 September 2022. Date of publication 25 October 2022; date of current version 3 November 2022. This letter was recommended for publication by Associate Editor T. P. Kucner and Editor M. Vincze upon evaluation of the reviewers' comments. This work was supported by the National Science Foundation under Grant 2224591. *(Corresponding author: Abubakar Siddique.)*

Abubakar Siddique is with the Department of Electrical and Computer Engineering, Marquette University, Milwaukee, WI 53233 USA (e-mail: abubakar.siddique@marquette.edu).

Amy Tabb is with the United States Department of Agriculture (USDA), Kearneysville, WV 25430 USA (e-mail: amy.tabb@gmail.com).

Henry Medeiros is with the Department of Agricultural and Biological Engineering, University of Florida, Gainesville, FL 32611 USA (e-mail: hmedeiros@ufl.edu).

Digital Object Identifier 10.1109/LRA.2022.3217000

II. Related Work

In agricultural automation, several supervised [13], [14], [15] and weakly supervised [16] deep learning models have been employed to address the challenges of detecting flowers [2], [3], [17], [18], fruits [4], [19], [20], or entire plants [21]. Applications of these methods range from robotic harvesting to estimating fruit load and optimizing fruit production by counting flowers in the early blooming season. Although some of these approaches leverage data augmentation techniques to generate automatic labels [12], [22], [23], none of these methods addresses model generalization ability for significantly different test datasets. In the context of object detection and segmentation, recent methods attempt to accommodate data distribution shifts through the following techniques: a) supervised learning, b) semi-supervised learning, c) self-supervised learning, and d) multi-task panoptic segmentation models.

a) Supervised Methods: These methods usually employ basic image transformations [10], [13] or sophisticated data augmentation techniques [24], [25] to improve model generalization. In addition to data augmentation during training, some methods incorporate post-processing algorithms at test time [26], [27] or include specialized input/output units that are easier to fine-tune to new datasets [28], [29]. While these techniques reduce the dependency on annotations for different datasets, they do not eliminate it. Model performance is still largely dependent on the amount of training data available.

b) Semi-supervised Methods: Using labeled data to bootstrap a model whose predictions are then employed to fine-tune the initial model (or to train a student model) is a popular approach to develop methods for multiple object detection [30], as well as instance [10], [12] and semantic [9] segmentation. This strategy is effective when labeled and unlabeled data have similar appearance and sufficient labeled data is available to bootstrap a deep model. When the characteristics of the labeled and unlabeled data differ significantly, as is the case among different flower species, more sophisticated supervision mechanisms are needed [31], [32].

c) Self-supervised Methods: When no labeled data is available, self-supervision strategies can be used to automatically generate pseudo-labels from the unlabeled data [33], [34]. In these scenarios, the initial model is trained to solve a surrogate task that presumably has a similar representation structure as the target task [35]. Using unsupervised learning techniques to align latent feature representations is a widely used approach [31]. Self-supervision strategies that use model prediction uncertainties to guide the learning process, while arguably more interpretable and predictable, are less commonly explored. Our approach uses a multi-inference data augmentation mechanism in conjunction with the region growing refinement (RGR) algorithm [26] to generate robust and accurate pseudo-labels in an iterative manner. These pseudo-labels allow our model to continuously improve its performance on previously unseen datasets.

d) Panoptic Methods: Multi-task learning is commonly used to improve model performance across different tasks [36]. As long as the tasks are similar, the model tends to generalize better to unseen data [37]. The recently introduced panoptic segmentation approach jointly learns the closely related tasks of instance and semantic segmentation and currently represents the state of the art in instance and semantic segmentation [38], [39]. However, training such models requires a significant number of manual labels containing instance and semantic information. Our approach makes it possible to apply a panoptic model to significantly different datasets without resorting to manual labels. To our knowledge no self-supervised panoptic segmentation method has been proposed so far.

Algorithm 1: Self-supervised Learning Algorithm.

Input: Set of high resolution labeled images I, their corresponding segmentation labels $\hat{I}$, and the set of unlabeled images I'

Output: Self-supervised model f^{W_r} for unlabeled data I'

1: Generate the augmented training set D_l using I and $\hat{I}$ according to (1)
2: Train the initial model $f^{W_0}(D_l)$ using D_l
3: Generate the augmented unlabeled image patches $Y_{\theta_{ij}}$
4: **for** $r \leftarrow 1$ to maxIter **do**
5: Generate the augmented predictions $\overline{Y}_{\theta_{ij}}$ using (3)
6: Compute the normalized score map $\mathcal{O}_i$ using (4)
7: Compute the binary semantic mask S_i from $\mathcal{O}_i$ using RGR
8: Generate the augmented binary semantic masks $S_{\theta_{ij}}$
9: Apply connected component analysis to $S_{\theta_{ij}}$ to find the instance masks $m^{(l)}_{\theta_{ij}}$ and bounding boxes $b^{(l)}_{\theta_{ij}}$
10: Construct the set of pseudo-labels $\widetilde{Y}_{\theta_{ij}}$ using (5)
11: Construct the set $D_u = \{Y_{\theta_{ij}}, \widetilde{Y}_{\theta_{ij}}\}$
12: Update the self-supervised model $f^{W_{r-1}}(D_u)$ using D_u
13: **end for**

III. Self-Supervised Panoptic Segmentation

Our proposed self-supervised learning (SSL) technique for panoptic segmentation shown in Fig. 1 comprises three main components: i) labeled and unlabeled data augmentation, ii) panoptic model initialization using the labeled dataset, and iii) panoptic pseudo-label generation from unlabeled data to update the model. As shown in Algorithm 1, we use images from the training set and their corresponding labels to train our initial model using an SL strategy. Our SSL approach then updates the initial model iteratively in a fully self-supervised manner using the pseudo-labels generated by the model at a previous iteration.

A. Data Augmentation

Our method is based on the panoptic segmentation model proposed in [11] pre-trained on the COCO [40] and COCO-stuff [41] datasets. To fine-tune the model for flower segmentation, we augment the training set introduced in [2] using a sliding window (SW) technique. That is, we extract from the input image I and its corresponding semantic label $\hat{I}$, both of size $M \times N$

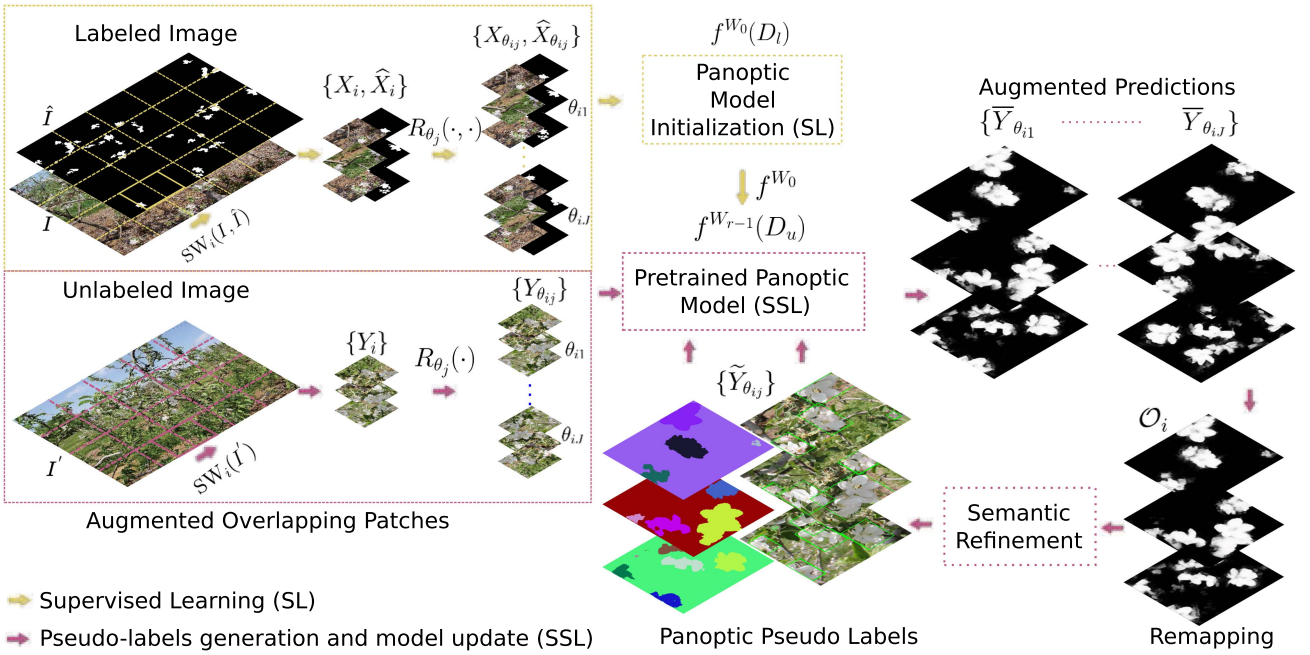

Fig. 1. Proposed self-supervised learning framework for multi-species flower segmentation. Labeled images are used to initialize the model for flower segmentation. The overlapping sliding window patches of the unlabeled input images are rotated multiple times to generate augmented semantic predictions from a previously initialized panoptic segmentation model. The remapping step transforms the score maps to the input coordinate system and then the normalized predictions are used to generate the panoptic pseudo-labels using a semantic refinement procedure to update the pre-trained model.

pixels, overlapping patches of size $m \times n = \lfloor M/K \rfloor \times \lfloor N/K \rfloor$ pixels with a stride of $p \times q = \lceil m/2 \rceil \times \lceil n/2 \rceil$, where K is the window size factor. Let $(X_i, \widehat{X}_i) = \mathrm{SW}_i(I, \hat{I})$ be the i-th image patch and its corresponding semantic label. We augment X_i and $\hat{X}_i$ by applying J different rotations at randomly selected angles $\{\theta_j\}_{j=1}^{J}$. For the sake of sampling efficiency, rather than directly sampling from the interval $[0, 2\pi]$, we employ a stratified sampling strategy. That is, we partition the circle into five sectors centered at $(\pi/2) \cdot k$, $k = 0, 1, \ldots, 4$ and sample each sector uniformly. This strategy increases sample diversity, ultimately reducing the variance of the pseudo-labels generated using our method. Thus, the set of labeled image patches and corresponding manual labels used to train the supervised model is given by

$$D_l = \left\{ \left(X_{\theta_{ij}}, \widehat{X}_{\theta_{ij}} \right) \right\} = \left\{ R_{\theta_j}(\mathrm{SW}_i(I, \hat{I})) \right\}, \qquad (1)$$

where $R_{\theta_j}(\cdot, \cdot)$ rotates its two arguments by an angle θ_j.

We employ the same data augmentation procedure for each unlabeled image of the test sets to generate the unlabeled augmented samples $Y_{\theta_{ij}}$ from the corresponding image patches Y_i. In the SSL approach, we use the SL model to predict the initial augmented pseudo-labels $\widetilde{Y}_{\theta_{ij}}$ used to fine-tune the model for unseen datasets. The procedure for pseudo-label generation is described in detail in Section III-B. Thus the unlabeled dataset for each flower species is

$$D_u = \left\{ \left(Y_{\theta_{ij}}, \widetilde{Y}_{\theta_{ij}} \right) \right\}. \qquad (2)$$

At test time, we simply apply the sliding window operation to generate the normalized semantic score maps and combine the predictions corresponding to the overlapping portions of each window using majority voting. We observed that the benefit of test-time data augmentation is negligible after a few SSL training iterations. Hence, we do not perform rotation augmentation at inference time, which ensures that the computational time of the model remains unchanged.

B. Pseudo-Label Generation

Data distribution shifts degrade the accuracy of segmentation models. Strong data augmentation is an effective strategy to mitigate this problem [42]. Thus, to improve the sensitivity of our model to different flower species, we apply the data augmentation procedure described above to Y_i and use the previously computed network weights $W_{(r-1)}$ to generate the augmented predictions at the r-th SSL iteration according to

$$\overline{Y}_{\theta_{ij}} = f^{W(r-1)}(Y_{\theta_{ij}}). \qquad (3)$$

To remap the semantic predictions back to the original image coordinate frame, we apply the reverse rotation operator $R_{-\theta_j}(\cdot)$ with bi-linear interpolation to the augmented predictions $\overline{Y}_{\theta_{ij}}$. We then normalize the scores using a softmax function and use the average normalized score map $\mathcal{O}_i$ as our final prediction, i.e.,

$$\mathcal{O}_i = \frac{1}{J} \sum_j \sigma \left(R_{-\theta_j}(\overline{Y}_{\theta_{ij}}) \right), \qquad (4)$$

where $\sigma(\cdot)$ represents the softmax function applied element-wise to the individual logits for the classes $\mathcal{C} \in \{\text{background, flower}\}$. As Fig. 2(a) and (b) illustrate, $\mathcal{O}_i$ contains a significantly higher

(a) $\bar{Y}_{\theta_{ij}}$ (b) $\mathcal{O}_i$ (c) $b_{\theta_{ij}}$ (d) $m_{\theta_{ij}}, S_{\theta_{ij}}$

Fig. 2. Illustration of the steps of our panoptic pseudo-label generation method. (a) semantic prediction for a single augmented patch, (b) normalized average score map obtained using (4), (c) instance bounding boxes, and (d) instance segmentation masks and semantic labels generated during SSL iterations.

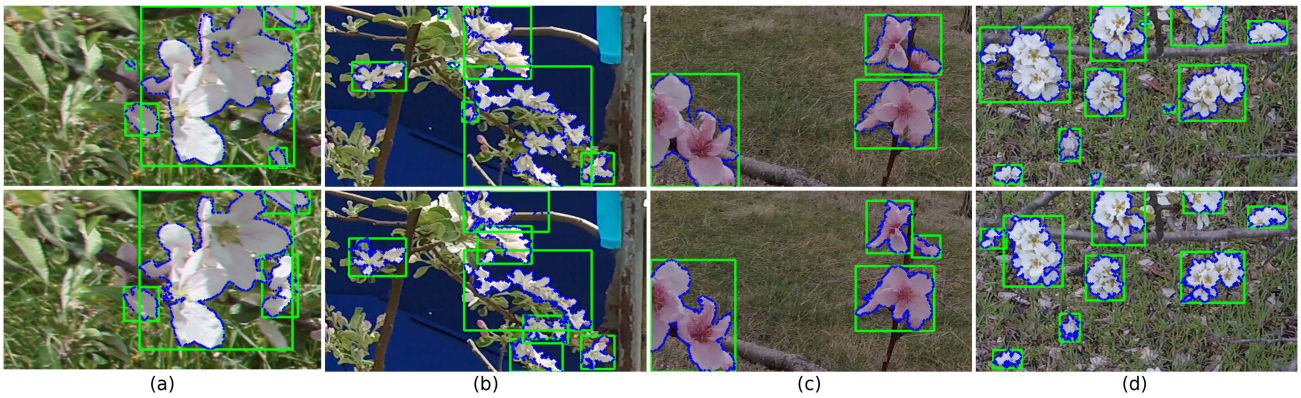

Fig. 3. Comparisons between the pseudo-labels generated using a fixed threshold τ_{seg} (top row) and the RGR-based semantic refinement (bottom row). (a) AppleA, (b) AppleB, (c) Peach, (d) Pear. The segmentation masks in the images at the bottom row better reflect flower boundaries and the corresponding bounding boxes better distinguish nearby flower instances.

number of flowers segmented with high confidence than a single augmented patch $\overline{Y}_{\theta_{ij}}$.

C. Semantic Prediction Refinement

Instead of applying a fixed threshold τ_{seg} to generate panoptic pseudo-labels from $\mathcal{O}_i$, we employ RGR, a robust segmentation refinement method [26]. RGR uses a Monte Carlo strategy to perform an appearance-based refinement of low-confidence regions in $\mathcal{O}_i$ using the corresponding image patch Y_i, which allows it to generate an improved binary segmentation mask. RGR uses three key elements to determine the boundaries of an object of interest: 1) the confidence of the model predictions, 2) appearance similarities among pixels, and 3) distances among pixels. That is, every pixel in an image is associated with a nearby pixel of similar appearance whose semantic class has been predicted with high confidence. As Fig. 3 illustrates, RGR improves the boundary adherence of the pseudo-labels and better distinguishes flower instances.

Let S_i be the semantic binary mask obtained from $\mathcal{O}_i$ using RGR. As in the pseudo-label generation step, we apply J rotations to S_i to generate augmented semantic binary masks, $S_{\theta_{ij}} = R_{\theta_j}(S_i)$. We then perform connected component analysis to obtain the corresponding instance masks $m^{(l)}_{\theta_{ij}}$ and bounding boxes $b^{(l)}_{\theta_{ij}}$ for the $l = 1, \ldots, L$ distinct elements of $S_{\theta_{ij}}$. The augmented panoptic pseudo-labels are given by

$$\widetilde{Y}_{\theta_{ij}} = \left\{ (b^{(l)}_{\theta_{ij}}, m^{(l)}_{\theta_{ij}}), S_{\theta_{ij}} \right\}_{l=1}^{L}. \tag{5}$$

Fig. 2(c) and (d) show that this approach generates high-quality bounding boxes and instance masks.

D. Multi-Task Loss

In both the SL and SSL models, the instance bounding boxes $b^{(l)}_{\theta_{ij}}$ and segmentation masks $m^{(l)}_{\theta_{ij}}$ from the augmented labels are used to train the ROI-heads for the flower class. The augmented semantic masks $S_{\theta_{ij}}$ are used to train the semantic segmentation head for the background and flower classes. For panoptic segmentation learning, we consider background as a stuff class and flower as a thing class [43] to jointly update the model using the following multi-task loss function

$$\mathcal{L}(W) = \lambda(\mathcal{L}_c + \mathcal{L}_b + \mathcal{L}_m) + (1 - \lambda)\mathcal{L}_s, \tag{6}$$

where $\mathcal{L}_c$ is the classification loss, $\mathcal{L}_b$ is the bounding-box loss, $\mathcal{L}_m$ is the mask loss, and $\mathcal{L}_s$ is the segmentation loss, as defined in [11]. By further training the initial SL model on the unlabeled datasets using the proposed SSL approach where the augmented panoptic labels are robust to prediction uncertainty and intrinsically incorporate rotation invariance, it is possible to iteratively improve the performance of the model.

Fig. 4. Examples of improved annotations in the AppleA training set. The cropped sections shows (a) incorrect contours containing background pixels, and (b) improved labels.

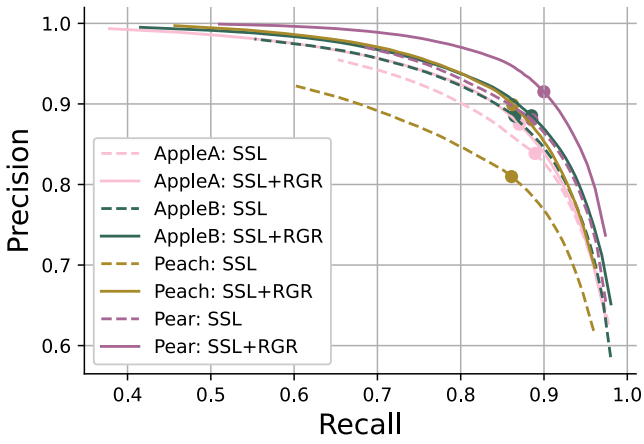

Fig. 5. Precision-recall curves for the SSL models with and without RGR pseudo-label refinement. Solid circles represent points that maximize F_1 scores.

IV. Experiments

We compare the performance of our method against the state-of-the-art algorithms presented in [2], [3] using the evaluation metrics and procedures described in [2]. To quantify the benefit of employing RGR as part of our pseudo-label generation strategy, we evaluate two different techniques to generate the pseudo-labels. First, we evaluate an approach in which we apply a fixed threshold τ_{seg} to the predicted score maps. For a fair comparison, we determine τ_{seg} based on the maximum F_1 score obtained by the model on the training set at a previous iteration (see Fig. 5). We call this model `SSL`. The model in which we employ RGR to refine the score maps without hard thresholding is deemed `SSL+RGR`. We also assess the performance improvements obtained by applying RGR as a post-processing mechanism in conjunction with our SSL model. We refer to that approach as `SSL+RGR (pp)`, where *pp* stands for post-processing. As a baseline, we also assess the performance of the `SL` model trained only on the AppleA dataset applied to the other datasets.

A. Datasets

We evaluate our method on the multi-species flower dataset first introduced in [2], which comprises four subsets: i) AppleA (train/test), ii) AppleB, iii) Peach, and iv) Pear. The AppleA and AppleB datasets contain images of the same apple orchard, but collected on different dates and under distinct conditions. While AppleA was collected using a hand-held camera, AppleB images were captured by a camera mounted to a mobile platform. For additional details regarding the datasets, we refer the reader to [2].

We train our SL model using the AppleA training set, which consists of 100 images with a resolution of $M \times N = 5184 \times 3456$ pixels [2]. After applying J rotation augmentation steps, the number of training patches $X_{\theta_{ij}}$ for each input image is $J \times (2K-1)^2$ since $i = 1, 2, \ldots, (2K-1) \times (2K-1)$ and $j = 1, 2, \ldots, J$. Hence, for $K = 4$ and $J = 20$, there are 98,000 training patches in the AppleA dataset. These patches are used to train our initial panoptic flower segmentation model.

We consider a randomly selected subset comprising 70% of the 30 images from the AppleA test set as unlabeled images I' to fine-tune the SL model using the automatically generated panoptic pseudo-labels. Similarly, 70% of the images from the AppleB, Peach, and Pear datasets (18, 24, and 18 images, respectively), all of which have a resolution of 2704×1520 pixels, are considered unlabeled images used to update the SL model iteratively. The remaining images in each dataset are used solely for performance evaluation. Given the relatively small size of the test sets, we evaluate our methods using five-fold cross-validation.

The datasets introduced in [2] provide pixel-level, high-resolution annotations of individual flowers. However, as Fig. 4 shows, the annotations have imperfections that can only be observed when closely inspected. Despite being small, these inaccuracies comprise a non-negligible portion of the image pixels, especially considering that only a fraction of the pixels correspond to flowers. To resolve this issue, we use the MATLAB® image labeler tool to manually correct inaccurate labels and to label additional smaller but clearly visible unannotated flowers. Fig. 4 shows some examples of the annotations before and after the corrections.

B. Training Details

The vast majority of image pixels in the datasets correspond to background pixels. Hence, to provide the model sufficient samples containing flower pixels, we train the network for 20,000 iterations using stochastic gradient descent with a batch size of 512 samples and a base learning rate of $25\mathrm{e}{-4}$, which is divided by 10 at 10%, 25%, and 50% of the training period. We freeze the ResNet-101 backbone [44] during training. To emphasize semantic learning, we use $\lambda = 0.8$ in (6). We have empirically observed that setting RGR's average spacing between samples to 100 pixels provides an adequate balance between the accuracy of the refined score map and the computation time required to produce it. We use the values reported in [2] for the remaining parameters, namely, the number of iterations is 10, the score map threshold is 0.5, the high-confidence foreground threshold is 0.8, and the high-confidence background threshold is 0.01.

V. Results and Discussion

Table I compares the performance of the SL and SSL models against the algorithms presented in [2], [3]. Although the SL model trained using our proposed data augmentation strategy segments flowers using a fixed threshold τ_{seg}, it performs either

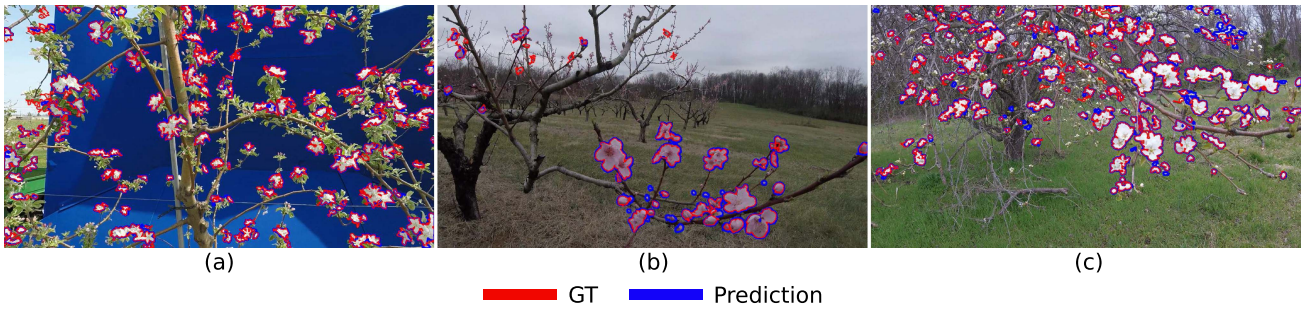

Fig. 6. Qualitative assessment of our proposed SSL approach on test datasets (a) AppleB, (b) Peach, (c) Pear. Most false positives correspond to small, unlabeled flowers.

TABLE I
EVALUATION OF FLOWER SEGMENTATION PERFORMANCE USING OUR SSL PANOPTIC MODEL

Dataset	Method	IoU	F1	Rcll	Prcn
AppleA	DeepLab+RGR [2]	71.4	83.3	87.7	79.4
	DeepLab+SCL [3]	**81.1**	**89.6**	**91.9**	87.3
	SL	77.1±0.9	87.0±0.5	86.7±0.6	87.3±0.8
	SSL	76.2±0.6	86.1±0.7	88.2±0.9	84.8±0.9
	SSL+RGR	77.9±0.6	87.5±0.3	87.8±0.6	87.3±0.6
	SSL+RGR (pp)	79.6±0.6	88.6±0.3	89.2±0.6	**88.1±0.7**
AppleB	DeepLab+RGR [2]	63.0	77.3	**91.2**	67.1
	DeepLab+SCL [3]	65.3	79.6	72.7	87.4
	SL	75.8±0.8	86.2±0.5	85.4±1.1	87.1±0.5
	SSL	76.8±0.7	86.8±0.4	87.0±0.7	86.7±0.8
	SSL+RGR	78.7±0.4	88.1±0.2	87.9±0.3	88.2±0.7
	SSL+RGR (pp)	**79.9±0.8**	**88.9±0.5**	86.7±1.0	**92.2±0.3**
Peach	DeepLab+RGR [2]	59.0	74.2	64.8	86.8
	DeepLab+SCL [3]	64.3	77.7	70.3	88.2
	SL	48.9±3.5	65.6±3.2	62.6±4.2	68.9±2.6
	SSL	67.8±4.1	80.7±2.9	**85.3±2.1**	76.7±3.6
	SSL+RGR	75.2±3.2	85.8±2.1	84.6±1.9	86.9±2.4
	SSL+RGR (pp)	**78.3±3.2**	**87.8±1.7**	84.9±2.1	**91.1±3.0**
Pear	DeepLab+RGR [2]	75.4	86.0	79.2	94.1
	DeepLab+SCL [3]	74.5	85.4	75.4	**97.3**
	SL	77.3±1.9	87.2±1.3	85.1±2.4	89.4±0.7
	SSL	78.6±1.7	87.9±1.0	87.9±1.6	88.1±0.8
	SSL+RGR	82.4±1.9	90.4±1.2	**89.4±1.8**	91.3±1.4
	SSL+RGR (pp)	**84.2±2.1**	**91.4±1.2**	87.4±1.9	95.8±0.9

The best results are shown in boldface and the second-best are underlined. We report the average value of the evaluation measures and their standard deviations across five runs.

TABLE II
PERFORMANCE IMPACT OF SLIDING WINDOW SIZE AND NUMBER OF ROTATION ANGLES

Dataset	$M \times N$	K	J	IoU	F_1	Inf. Time (sec.)
AppleA	5184 × 3456	4	1	73.6	84.8	7.2
		8		75.4	86.0	15.4
		16		53.3	69.5	90.0
AppleB	2704 × 1520	2	1	71.6	83.0	1.4
		4		76.7	86.8	5.3
		8		57.1	72.6	22.1
Peach	2704 × 1520	4	1	51.3	67.8	5.5
			5	58.2	73.6	34.7
			10	60.3	75.2	71.5
			20	61.3	76.0	147.8

on par with or better than the state-of-the-art models on test sets that are similar to the training set, even without applying our proposed SSL strategy. However, for datasets with significantly different characteristics, the SL model does not perform satisfactorily. The SSL approach using a fixed threshold outperforms the baseline methods on the AppleB, Peach, and Pear datasets by significant margins (11.5%, 3.5%, and 4.1% absolute IoU improvement with respect to [3]). For the AppleA dataset, the SSL method alone outperforms [2] but is slightly worse than [3]. This is largely due to the fact that the baseline methods perform dramatically better on the training set, whereas the performance of our model remains relatively stable across datasets. As discussed in more detail below, background flowers also contribute to the performance degradation. When we use RGR to refine the pseudo-labels, we observe IoU improvements with respect to the fixed threshold SSL method of 1.9%, 7.4%, and 3.8% for the AppleB, Peach, and Pear datasets, respectively. The performance improvements obtained with RGR are proportional to the appearance dissimilarities between the AppleA dataset used for model pre-training and the corresponding target dataset. The average hue, saturation, and value difference between the AppleA dataset and the AppleB dataset is 30.3, whereas for the Peach and Pear datasets it is 76.9 and 28.9, respectively. Finally, performing an additional RGR step at test time leads to an additional average IoU improvement of approximately 1.9% but at the cost of substantially higher inference times, as discussed in the next section. Fig. 5 shows the precision-recall curves for the proposed SSL methods with and without pseudo-label refinement using RGR.

The qualitative results in Fig. 6 show that the SSL models are highly sensitive to flowers in complex regions. For some datasets, the SSL methods show slightly lower precision than [3]. The main reason for the lower precision is the presence of small, unannotated flowers in the datasets that our model can detect. This can be observed in Fig. 6(c) where several small flowers are present, especially on branches farther from the camera. Determining which flowers should be annotated is an application-specific problem that requires further investigation.

A. *Parameter Sensitivity and Computation Time Analysis*

Table II shows the impact of the sliding window size factor K and the number of rotation angles J on model performance and average inference time per input image. This evaluation is performed on the first SSL iteration of a model initialized with $K = 4$ and $J = 20$. That is, the evaluation reflects the

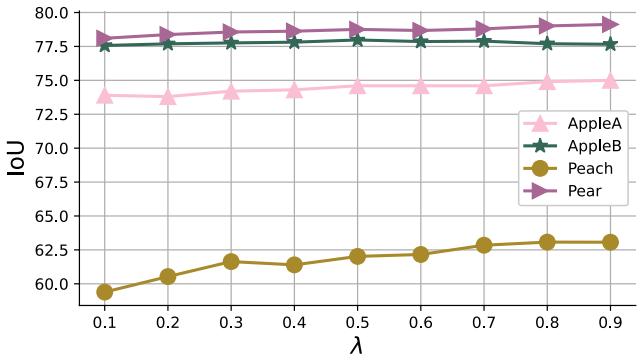

Fig. 7. Impact of the loss weight λ (6) on flower segmentation performance at the first SSL iteration with $J = 20$ and $K = 4$.

impact of model parameters on the accuracy of the resulting pseudo-labels. The top two rows show the test-time impact of varying K without employing test-time rotations (i.e., $J = 1$) for the AppleA and AppleB datasets, respectively. The last row of the table shows that the IoU and F_1 measures on the Peach dataset gradually increase with J when rotation augmentation is employed at inference time, but so does the computation time. Inference times were obtained using one NVIDIA ® GeForce® RTX 2080 Ti GPU without any multi-processing technique. Post-processing times using RGR are approximately $16\times$ higher than those presented in Table II on our Intel® Xeon® Silver 4112 CPU @2.6 G Hz. Results for the remaining datasets are similar and are omitted for brevity. Fig. 7 shows the impact of λ in the multi-task loss (6) for different flower species. Although the performance of our approach remains relatively stable as we vary λ, for most datasets, the best results are obtained with $0.7 \leq \lambda \leq 0.9$, especially in cross-species scenarios, where appearance variation is more prominent.

VI. Conclusion

We introduced a self-supervised learning technique to accurately segment multiple tree flower species without significant manual labeling efforts. To automatically generate instance and semantic labels for unlabeled datasets, we propose a data augmentation technique associated with a semantic segmentation refinement strategy that produces accurate pseudo-labels for self-supervised model training. The proposed SSL technique makes it possible to train a deep multi-task model effectively on unlabeled fruit flower datasets. Self-supervised learning substantially reduces model dependency on computationally expensive post-processing steps to further refine the model predictions at inference time. That being said, employing a post-processing approach with our SSL model can further improve its prediction accuracy. Our novel SSL method creates a new baseline for the multi-species flower segmentation task.

A robust and accurate multi-species flower detection method is the first step toward the development of autonomous robotic thinning systems [45]. In the future, the proposed panoptic flower segmentation algorithm can be further improved in a number of ways. First, our proposed framework resorts primarily to a data augmentation strategy based on image rotations. Given the characteristics of the problem under consideration, it stands to reason that additional data augmentation strategies such as color jittering and image blurring would further contribute to the generation of accurate pseudo-labels. In addition, instead of using empirically defined weights for the instance and semantic segmentation tasks, task-dependent uncertainty learning strategies [46] may better capture appearance variations to optimize the task weights. Finally, pseudo-label pixels or sometimes entire instances may have low prediction scores. The uncertainty of the pseudo-labels may be used to weigh the contributions of individual samples. Uncertainty-weighed loss functions [12] are a promising technique to accomplish that goal.

References

[1] G. Farjon, O. Krikeb, A. B. Hillel, and V. Alchanatis, "Detection and counting of flowers on apple trees for better chemical thinning decisions," *Precis. Agriculture*, vol. 21, no. 3, pp. 503–521, 2020.
[2] P. A. Dias, A. Tabb, and H. Medeiros, "Multispecies fruit flower detection using a refined semantic segmentation network," *IEEE Robot. Automat. Lett.*, vol. 3, no. 4, pp. 3003–3010, Oct. 2018.
[3] K. Sun, X. Wang, S. Liu, and C. Liu, "Apple, peach, and pear flower detection using semantic segmentation network and shape constraint level set," *Comput. Electron. Agriculture*, vol. 185, 2021, Art. no. 106150.
[4] P. Akiva, K. Dana, P. Oudemans, and M. Mars, "Finding berries: Segmentation and counting of cranberries using point supervision and shape priors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2020, pp. 50–51.
[5] U. Bhattarai and M. Karkee, "A weakly-supervised approach for flower/fruit counting in apple orchards," *Comput. Ind.*, vol. 138, 2022, Art. no. 103635.
[6] G. Wang, W. Li, M. Aertsen, J. Deprest, S. Ourselin, and T. Vercauteren, "Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks," *Neurocomputing*, vol. 338, pp. 34–45, 2019.
[7] C. Luo, Y. Zhu, L. Jin, and Y. Wang, "Learn to augment: Joint data augmentation and network optimization for text recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 13746–13755.
[8] W. H. Maes and K. Steppe, "Perspectives for remote sensing with unmanned aerial vehicles in precision agriculture," *Trends Plant Sci.*, vol. 24, no. 2, pp. 152–164, 2019.
[9] S. A. Golestaneh and K. M. Kitani, "Importance of self-consistency in active learning for semantic segmentation," in *Proc. Brit. Mach. Vis. Conf.*, 2020.
[10] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8759–8768.
[11] A. Kirillov, R. Girshick, K. He, and P. Dollár, "Panoptic feature pyramid networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 6399–6408.
[12] A. Siddique and H. Medeiros, "Tracking passengers and baggage items using multi-camera systems at security checkpoints," 2020, *arXiv:2007.07924*.
[13] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 801–818.
[14] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.
[15] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2961–2969.
[16] I. H. Laradji, D. Vázquez, and M. Schmidt, "Where are the masks: Instance segmentation with image-level supervision," in *Proc. Brit. Mach. Vis. Conf.*, 2019.
[17] X. A. Wang, J. Tang, and M. Whitty, "Side-view apple flower mapping using edge-based fully convolutional networks for variable rate chemical thinning," *Comput. Electron. Agriculture*, vol. 178, 2020, Art. no. 105673.
[18] D. Wu, S. Lv, M. Jiang, and H. Song, "Using channel pruning-based YOLOv4 deep learning algorithm for the real-time and accurate detection of apple flowers in natural environments," *Comput. Electron. Agriculture*, vol. 178, 2020, Art. no. 105742.

[19] G. Li et al., "Real-time detection of kiwifruit flower and bud simultaneously in orchard using YOLOv4 for robotic pollination," *Comput. Electron. Agriculture*, vol. 193, 2022, Art. no. 106641.
[20] A. Koirala, K. B. Walsh, Z. Wang, and C. McCarthy, "Deep learning for real-time fruit detection and orchard fruit load estimation: Benchmarking of 'MangoYOLO'," *Precis. Agriculture*, vol. 20, pp. 1107–1135, Dec. 2019.
[21] D. Riehle, D. Reiser, and H. W. Griepentrog, "Robust index-based semantic plant/background segmentation for RGB- images," *Comput. Electron. Agriculture*, vol. 169, 2020, Art. no. 105201.
[22] S. Bargoti and J. Underwood, "Deep fruit detection in orchards," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2017, pp. 3626–3633.
[23] H. Kang and C. Chen, "Fast implementation of real-time fruit detection in apple orchards using deep learning," *Comput. Electron. Agriculture*, vol. 168, 2020, Art. no. 105108.
[24] E. D. Cubuk, B. Zoph, D. Mané, V. Vasudevan, and Q. V. Le, "Autoaugment: Learning augmentation strategies from data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 113–123.
[25] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," 2017, *arXiv:1712.04621*.
[26] P. A. Dias and H. Medeiros, "Semantic segmentation refinement by monte carlo region growing of high confidence detections," in *Proc. Asian Conf. Comput. Vis.*, 2018, pp. 131–146.
[27] C. Tang, H. Chen, X. Li, J. Li, Z. Zhang, and X. Hu, "Look closer to segment better: Boundary patch refinement for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 13926–13935.
[28] A. Kirillov, Y. Wu, K. He, and R. Girshick, "Pointrend: Image segmentation as rendering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9799–9808.
[29] M. Teichmann and R. Cipolla, "Convolutional CRFs for semantic segmentation," in *Proc. Brit. Mach. Vis. Conf.*, 2020.
[30] M. Xu et al., "End-to-end semi-supervised object detection with soft teacher," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 3060–3069.
[31] K. Saito, D. Kim, S. Sclaroff, and K. Saenko, "Universal domain adaptation through self-supervision," in *Proc. Neural Inf. Process. Syst.*, 2020, pp. 16282–16292.
[32] M. R. Vyas, H. Venkateswara, and S. Panchanathan, "Leveraging seen and unseen semantic relationships for generative zero-shot learning," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 70–86.
[33] W. Lee, J. Na, and G. Kim, "Multi-task self-supervised object detection via recycling of bounding box annotations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4984–4993.
[34] A. Newell and J. Deng, "How useful is self-supervised pretraining for visual tasks?," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 7345–7354.
[35] M. Larsson, E. Stenborg, C. Toft, L. Hammarstrand, T. Sattler, and F. Kahl, "Fine-grained segmentation networks: Self-supervised segmentation for improved long-term visual localization," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 31–41.
[36] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7482–7491.
[37] B. Cheng, G. Liu, J. Wang, Z. Huang, and S. Yan, "Multi-task low-rank affinity pursuit for image segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2439–2446.
[38] R. Mohan and A. Valada, "EfficientPS: Efficient panoptic segmentation," *Int. J. Comput. Vis.*, vol. 129, no. 5, pp. 1551–1579, 2021.
[39] Z. Li et al., "Panoptic SegFormer: Delving deeper into panoptic segmentation with transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 1280–1289.
[40] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.
[41] H. Caesar, J. Uijlings, and V. Ferrari, "COCO-Stuff: Thing and stuff classes in context," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1209–1218.
[42] J. Yuan, Y. Liu, C. Shen, Z. Wang, and H. Li, "A simple baseline for semi-supervised semantic segmentation with strong data augmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 8229–8238.
[43] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár, "Panoptic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9404–9413.
[44] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[45] M. Karkee, Q. Zhang, and A. Silwal, "Agricultural robots for precision agricultural tasks in tree fruit orchards," in *Proc. Innov. Agricultural Robot. Precis. Agriculture*, 2021, pp. 63–89.
[46] A. Siddique, R. J. Mozhdehi, and H. Medeiros, "Unsupervised spatio-temporal latent feature clustering for multiple-object tracking and segmentation," in *Proc. Brit. Mach. Vis. Conf.*, 2021.