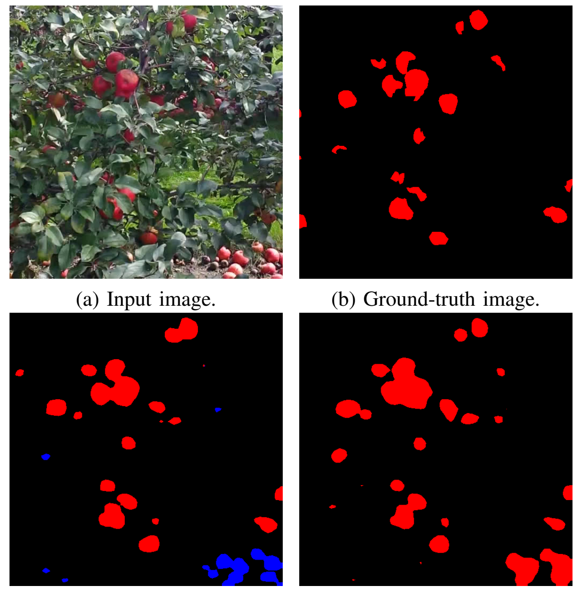
2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Few-Shot Fruit Segmentation via Transfer Learning

Jordan A. James[1], Heather K. Manching[2], Amanda M. Hulse-Kemp[2,3], and William J. Beksi[1]

***Abstract*— Advancements in machine learning, computer vision, and robotics have paved the way for transformative solutions in various domains, particularly in agriculture. For example, accurate identification and segmentation of fruits from field images plays a crucial role in automating jobs such as harvesting, disease detection, and yield estimation. However, achieving robust and precise infield fruit segmentation remains a challenging task since large amounts of labeled data are required to handle variations in fruit size, shape, color, and occlusion. In this paper, we develop a few-shot semantic segmentation framework for infield fruits using transfer learning. Concretely, our work is aimed at addressing agricultural domains that lack publicly available labeled data. Motivated by similar success in urban scene parsing, we propose specialized pre-training using a public benchmark dataset for fruit transfer learning. By leveraging pre-trained neural networks, accurate semantic segmentation of fruit in the field is achieved with only a few labeled images. Furthermore, we show that models with pre-training learn to distinguish between fruit still on the trees and fruit that have fallen on the ground, and they can effectively transfer the knowledge to the target fruit dataset.**

***Index Terms*— Agricultural Automation; Computer Vision for Automation; Object Detection, Segmentation and Categorization**



(a) Input image. (b) Ground-truth image.

(c) Two-shot CitDet results. (d) Two-shot ImageNet results.

Fig. 1: Qualitative results of our two-shot apple segmentation with specialized pre-training on CitDet [2] and generalized pre-training on ImageNet [3]. Predicted fruit on the tree are colored red while predicted fruit on the ground are colored blue. Models pre-trained on CitDet are capable of distinguishing between fruit on the ground and fruit on the tree using only small amounts of labeled data to learn from.

I. Introduction

Automation, principally in the realm of infield fruit segmentation, has ushered in a new era of modern agriculture. The integration of advanced technologies such as machine learning, computer vision, and robotics has started to revolutionize the way we cultivate, harvest, and process fruits. Through automated systems capable of accurately identifying and categorizing fruits while still on the plant, efficiency has surged resulting in reduced costs and minimized waste [1]. Nonetheless, despite these significant strides, the task of infield fruit segmentation still remains a complex challenge. The immense diversity of fruit shapes, sizes, colors, and textures poses difficulties in developing universally applicable algorithms. Moreover, the scarcity of labeled training data hinders the training of machine learning models, as collecting and annotating large datasets for every fruit variety is a resource-intensive endeavor.

Few-shot learning (FSL) has emerged as a contemporary research area that allows models to learn novel tasks from a few samples [4], [5]. In traditional machine learning, models require substantial amounts of data to achieve satisfactory performance. Conversely, FSL aims to enable models to generalize effectively even when presented with just a handful of examples. Transfer learning, a technique commonly employed in FSL, plays a pivotal role in this process. It involves training a model on a source task with ample data and then transferring the learned knowledge to a target task with limited data. This enables the model to leverage its understanding of the source task to bootstrap its performance on the target task. By pre-training on a large dataset and fine-tuning on the task of interest, transfer learning minimizes the need for vast amounts of labeled data. It accelerates the learning process for unique tasks, thus making it a crucial approach in advancing the capabilities of machine learning models in scenarios with limited data availability.

In this work, we show that accurate semantic segmentation of in-orchard apples can be achieved with scarce amounts of data using FSL with transfer learning, Fig. 1. Our method leverages a three-branch decoder for direct learning of fruit boundaries and pre-training on CitDet [2], an open-source annotated image dataset of infield citrus fruit. The network

[1]The authors are with the Department of Computer Science and Engineering, The University of Texas at Arlington, Arlington, TX, USA. Emails: jaj9608@mavs.uta.edu, william.beksi@uta.edu.

[2]The author is with the Department of Crop and Soil Sciences, North Carolina State University, Raleigh, NC, USA. Email: hkmanchi@ncsu.edu.

[3]The author is with the Genomics and Bioinformatics Research Unit, USDA Agricultural Research Service, Raleigh, NC, USA. Email: amanda.hulse-kemp@usda.gov.

is evaluated on the MinneApple [6] test set, which consists of 331 labeled images of in-orchard apples. In summary, our contributions are the following.

- A light-weight three-branch decoder that effectively learns task specific boundaries enabling knowledge transfer of learned shapes and boundaries between tasks.
- A framework for few-shot fruit segmentation that leverages specialized pre-training of a neural network and is designed to learn known features with similarities between fruits of different species such as the shape.

The source code associated with this project is publicly available at [7].

The remainder of the paper is organized as follows. We provide an overview of related research in Section II. The details of our framework are presented in Section III. Our evaluation results are discussed in Section IV. In Section V, we conclude and provide directions for future work.

II. Related Work

Representative work towards FSL, fruit detection and segmentation, and real-time semantic segmentation are discussed separately in this section.

A. Few-Shot Learning

ImageNet [3], an important benchmark dataset for computer vision, contains a vast collection of labeled images across numerous categories. Pre-training networks on ImageNet has proven to be an effective way of initializing weights for a target task with a small sample size [8]. Multiple works have shown that classifiers with a higher ImageNet accuracy achieve greater overall object detection and transfer accuracy [9], [10]. Furthermore, Hendrycks et al. [11] showed that pre-training improves model robustness and uncertainty. Huh et al. [12] also presented the advantages of ImageNet pre-training even with the removal of classes related to the target task.

Although pre-training on ImageNet has many benefits, pre-training on specialized datasets with high similarity to the target dataset can also improve accuracy. One notable domain where this has been used to achieve state-of-the-art results is urban scene parsing, which involves pre-training on the large Mapillary Vistas [13] dataset and fine-tuning on Cityscapes [14]. Specialized pre-training has been shown to benefit the accuracy of semantic segmentation for various neural network architectures [15], [16], [17]. In our work we show that specialized pre-training also benefits the task of infield fruit segmentation.

B. Fruit Detection and Segmentation

Infield detection and segmentation of fruit is vital for agricultural tasks including fruit yield, yield loss estimation, and automated picking. However, variable lighting conditions, occlusions, and clustering make the accurate recognition of fruit in an orchard environment very difficult. Over the last two decades, many machine vision systems have been developed to address these problems. Early techniques (e.g., [18], [19], [20]) for automating fruit recognition relied on hand-crafted features for specific fruits in order to exploit unique attributes such as color, texture, and shape to separate fruit from background foliage.

More recently, methods utilizing deep learning have shown great promise in the domain of fruit segmentation. For example, a fully convolutional neural network [21] was utilized by Bargoti and Underwood [22], followed by watershed segmentation and a circular Hough transform, to detect in-orchard apple instances from segmentation masks for yield estimation. Semantic segmentation was employed by Chen et al. [23] to detect and segment fruit blobs, which were then used as input to a convolutional neural network for yield estimation. In addition, Liu et al. [24] utilized a fully convolutional neural network to segment and detect fruits in a tracking pipeline.

The cost-efficient generation of synthetic data is also playing an increasing important role in deep learning for agriculture [25]. For instance, Rahnemoonfar and Sheppard [26] demonstrated that synthetic data can be used to train deep neural networks for counting tomatoes in the field. In addition, Barth et al. [27] used synthetically generated data to train a DeepLab [28] model to segment pepper images. Our approach distinguishes itself from prior research efforts by harnessing acquired knowledge in fruit image segmentation, specifically by employing knowledge gained from segmenting one fruit to enhance the segmentation of another distinct fruit.

C. Real-Time Semantic Segmentation

Real-time semantic segmentation algorithms are necessary for practical applications in agricultural automation that require fast interactions and responses. Many methods adopt an encoder-decoder architecture such as SwiftNet [29] and DFANet [30]. Notably, SwiftNet provides information to a light-weight decoder from two separate branches each processing feature maps at different resolutions. ShuffleSeg [31] proposes a decoder for ShuffleNet [32] with skip connections and 1×1 convolutions for channel transformations. The DeepLab [28] architecture proposes atrous convolutions, an atrous spatial pyramid pooling module for the decoder, and it adopts a high-resolution skip connection. FANet [33] modifies the traditional U-shape architecture with fast attention in between the skip links. It uses a light-weight ResNet-18 [34] backbone to attain real-time performance.

Recent works have achieved real-time performance using parallel branched networks. For instance, BiSeNetV1 [35] and BiSeNetV2 [36] employ a two-branch architecture consisting of a contextual detail branch and a spatial detail branch. This architecture was further enhanced by adding connections between branches in DDRNet [37], and a deep aggregation pyramid pooling module (DAPPM) was added to efficiently aggregate contextual details from the context branch. PIDNet [38] proposed a three-branch network, which utilizes an auxiliary derivative branch (ADB) to detect boundaries and guide the fusion between two other branches using a boundary attention guided (BAG) fusion module. Further experiments showed the benefit of adding the ADB-BAG method to previously proposed two-branch architectures. Finally, a parallel aggregation pyramid pooling module (PAPPM) inspired by the DAPPM was proposed for fast aggregation of contexts.

III. Few-Shot Fruit Segmentation

Our proposed framework is designed for fruit segmentation tasks with a scarce amount of available labeled data. We take a simple approach, consisting of pre-training on a similar task and standard data augmentation, to achieve accurate few-shot segmentation. To further facilitate the task transfer by learning the shape of the fruit, we use a three-branch decoder inspired by the design of PIDNet [38]. ResNet-18 [34] is adopted as the backbone encoder for a light-weight design with 23 million trainable parameters. The network architecture is depicted in Fig. 2.

A. Three-Branch Decoder

The decoder consists of a high-resolution spatial branch, a low-resolution context branch, and an ADB. The feature maps are fused using a BAG fusion module and then passed to a segmentation head to obtain the output mask.

1) Spatial Branch: For a light-weight design, the spatial branch relies heavily on the spatial representations learned by the backbone network. The channels of the feature map of the backbone with an output stride of 8 are transformed to match the channels of the ADB and context branch with just two convolutional layers.

2) Context Branch: The final feature map of the backbone is used as input for the context branch. Following the success of PIDNet [38] and DDRNet [37], we use a PAPPM at an output stride of 64 for efficient aggregation of contexts. A bottleneck block is included before the PAPPM to down-sample the input feature map to the output stride of 64 for improved computational efficiency.

3) ADB-BAG: The ADB and BAG module are incorporated to enhance the knowledge transfer between datasets by learning shapes and boundaries that are common between them. To learn boundaries, the second stage of the ADB is inputted to an auxiliary head and trained with binary cross-entropy loss (also denoted as the boundary loss) on generated boundary labels. The boundary masks are obtained by using the Canny edge detector [39] on the instance masks followed by dilation. The BAG module then fuses the contextual and spatial information along the learned boundaries by utilizing a sigmoid attention function.

B. Knowledge Transfer from Specialized Tasks

When dealing with specialized tasks or domains where collecting a large amount of labeled data is challenging, transfer learning can be highly advantageous. Instead of starting with a generic pre-trained model, the model is initialized with weights from a network that was pre-trained on a dataset similar to the target task. In this work, semantic segmentation of infield citrus is transferred to the task of few-shot semantic segmentation of in-orchard apples.

TABLE I: Quantitative results on the MinneApple test set with CitDet pre-training.

Pre-training	# of Training Images	mIoU	PA
CitDet	0	67.2	97.3
CitDet	2	76.7	98.0
CitDet	4	75.1	98.1
CitDet	670	82.5	98.7

TABLE II: Quantitative results on the MinneApple test set with ImageNet pre-training.

Pre-training	# of Training Images	mIoU	PA
ImageNet	2	66.7	95.8
ImageNet	4	69.5	96.9
ImageNet	670	82.1	98.7

1) Datasets: The CitDet [2] dataset is used for pre-training the model for the task of citrus segmentation. CitDet contains many high-resolution images of citrus trees with bounding box annotations for both fruit on the ground and on the trees. To obtain masks for pre-training a semantic segmentation model, we make use of Meta's Segment Anything Model [40] along with the prompt encoder to obtain high-quality instance masks for objects within the specified labels. The obtained instance masks are then combined and assigned the class label according to the original bounding box annotation. The model is then trained on images from CitDet resized to 1024×1024 with the obtained masks and the best model weights, determined by the highest validation mean intersection over union (mIoU), are saved for transfer learning.

MinneApple [6] is chosen as the target dataset for semantic segmentation of in-orchard apples. The dataset shares similarities with CitDet. For instance, both datasets contain images of entire trees and labels of fruit under varying amounts of occlusion. Yet, a key difference between the two datasets is that the fruit on the ground class is not included in MinneApple. Although the fruit on the ground class has no labels in the target dataset, we include the class as an output to allow a direct transfer of the previously learned distinctions between fruit on the ground and fruit on the tree.

2) Data Augmentation: To enable training with common batch sizes using sparse amounts of labeled data, we apply data augmentation techniques before and during training. Before training each image, a mask is scaled by [0.75, 1.0, 1.25, 1.5] and 5 random crops of size 512×512 are taken from each scale to produce 20 training images from each original image. Example visualizations of the random crops at each scale can be seen in Fig. 3. At training time, random horizontal flipping is applied to prevent the model from learning to predict fruit on only one side of an image.

IV. Evaluation

In this section, we experimentally assess our method on few-shot semantic segmentation of in-orchard apples using two and four training images, and evaluate on zero-shot transfer using CitDet pre-training. The class-wise mIoU and

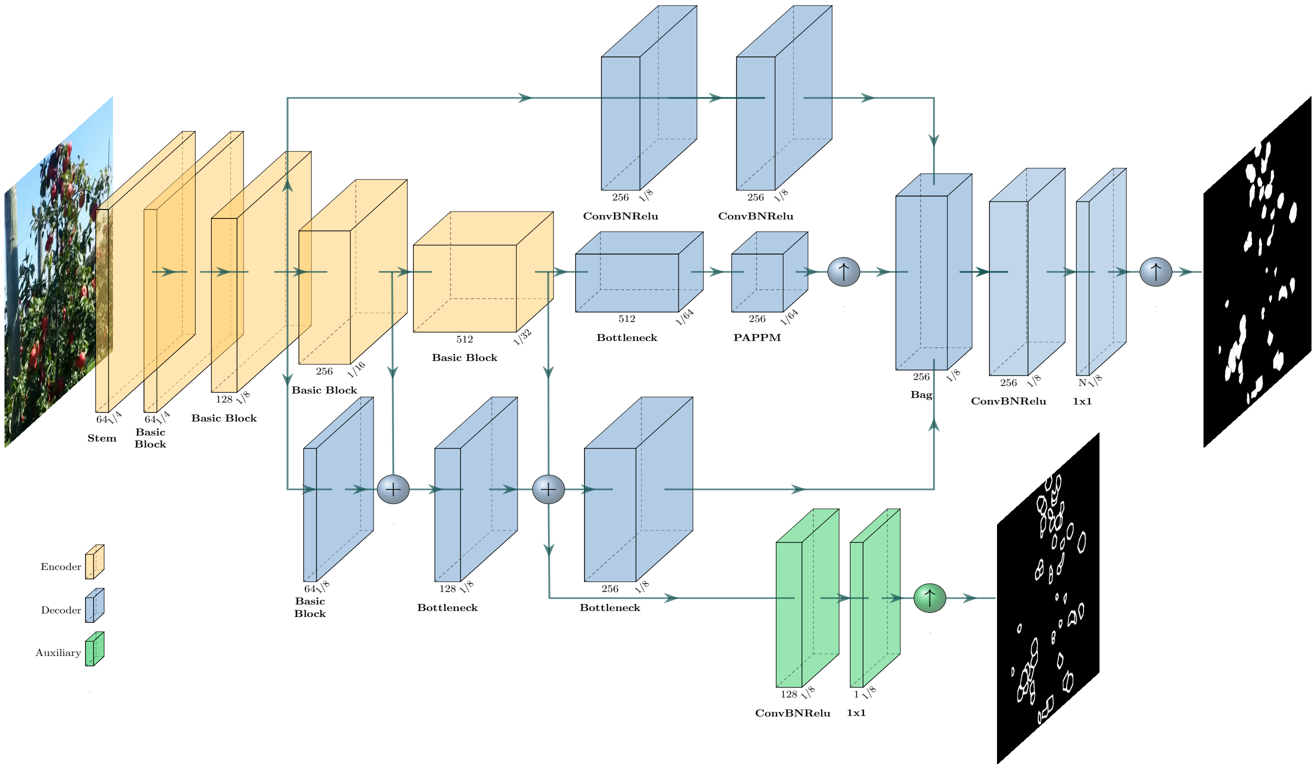


Fig. 2: The architecture of our few-shot fruit segmentation network. The encoder, ResNet-18, is colored yellow, the decoder is blue, and the auxiliary boundary head (train only) is green. The "Basic Block" is the basic residual block in ResNet-18, and the "Bottleneck" is the bottleneck block from ResNet-50. The ConvBNRelu blocks use 3 $\times$ 3 kernels for the convolutional layers.

(a) Input image. (b) Augmented input images.

Fig. 3: A visualization of the data augmentation used for few-shot semantic segmentation of in-orchard apples. The original image is shown in (a) and the resulting augmented image crops are shown in (b). Each panel in (b) displays a random crop of the scale used where the top left is scaled by 0.75, top right by 1.0, bottom left by 1.25, and bottom right by 1.5.

TABLE III: Quantitative results on the MinneApple test set with no pre-training.

Pre-training	# of Training Images	mIoU	PA
None	2	64.3	96.1
None	4	67.9	96.8
None	670	78.6	98.3

the percent of pixels that are accurately classified in the image, i.e., pixel accuracy (PA), are chosen as the evaluation metrics. All experiments were evaluated on the MinneApple test set containing 331 images. For all models, the fruit on the ground class was set to the background before evaluation.

The quantitative results on the MinneApple test set are shown in the Table I. We describe the implementation details and compare our results with typical full-shot semantic segmentation in Section IV-A. Additionally, we conduct experiments using traditional ImageNet pre-training as well as no pre-training with randomly initialized weights and report the results in Table II and Table III, respectively. Section IV-B provides an ablation study. The evaluation results and key findings are discussed in Section IV-C.

A. *Implementation Details*

Our implementation was developed using PyTorch and the Lightning [41] package. We utilized the timm open-source library to import the pre-trained weights from the ImageNet dataset for the ResNet encoder. For all methods and datasets, a "poly" learning rate scheduled with a factor of 0.9 was used. The stochastic gradient descent optimizer with a weight decay of 1E-4 and momentum of 0.9 was employed with a batch size of 4 for fine-tuning. The proposed data augmentation was performed before training and the same random crops were used for all experiments.

Hyperparameters, such as the number of epochs and start-

(a) Input image. (b) Ground-truth image. (c) Prediction images.

Fig. 4: Qualitative results for few-shot and full-shot apple segmentation for each pre-training method. Predicted fruit on the tree are colored cyan and predicted fruit on the ground are colored magenta. The original input can be seen in (a), the ground-truth labels in (b), and predictions in (c). Each row of predictions correspond to the pre-training method. The top row is CitDet, the middle row is ImageNet, and the bottom row is no pre-training. Each column corresponds to the number of training images used for fine tuning (i.e., 2, 4, and 670), increasing from left to right (best viewed zoomed in).

TABLE IV: Ablation study results for a two-branch architecture.

Pre-training	# of Training Images	mIoU	PA
CitDet	0	63.8	95.3
ImageNet	2	60.4	92.7
CitDet	2	71.6	95.7
ImageNet	4	65.0	94.9
CitDet	4	72.8	95.7
ImageNet	670	73.4	96.3
CitDet	670	73.9	95.9

ing learning rate, were held constant between experiments with a varying number of training images. In all experiments the model with the best performance on the test set was saved. For models initialized with random weights and no pre-training, the starting learning rate was set to 7.5E-3 and each model was trained for 100 epochs. For experiments with ImageNet, the pre-trained encoder weights were imported from timm and the entire network was fine-tuned at a starting learning rate of 1E-3 for 50 epochs. Finally, transfer from CitDet was achieved by importing pre-trained weights for both the encoder and decoder, and fine-tuned with a starting learning rate of 1E-4 for 50 epochs.

B. Ablation Study

To validate the design of the three-branch decoder and separate the effects of our specialized pre-training from the effects of the architecture, we performed an ablation study on the ADB and corresponding BAG fusion module. Specifically, we removed the ADB and replaced the BAG fusion module with the addition of the spatial and context branches to create a two-branch decoder. The same training and pre-training methods were adopted for the two-branch decoder. The models were then evaluated on the MinneApple test set using the mIoU and PA metrics as in the previous experiments. The results in Table IV demonstrate that pre-training on CitDet also greatly benefits few-shot semantic segmentation of in-orchard apples for architectures without the ADB-BAG modules. Comparing the quantitative results of the two-branch and three-branch architectures, the inclusion of the ADB-BAG modules shows a large increase in performance for both few-shot and full-shot segmentation.

C. Discussion

We observe from the qualitative visualizations in Fig. 1 and Fig. 4 that few-shot models without specialized pre-training were capable of segmenting nearly all apples from the image. The main shortcoming of these models compared to those pre-trained on CitDet is the inability to distinguish the difference between labeled fruit on the trees and unlabeled fruit on the ground. This could be a problem for applications requiring low latency such as fruit picking, as post-processing with traditional computer vision methods would need to be used to distinguish between the two classes. For all the pre-training methods, we noticed that the boundaries of the masks became more refined as more

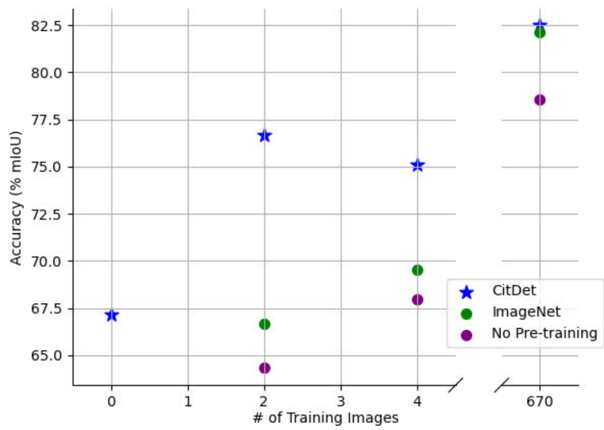
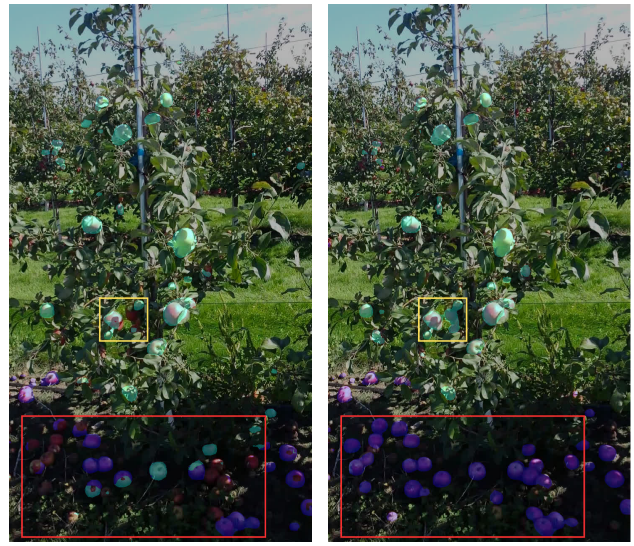


Fig. 5: The mIoU performance on the MinneApple test set versus the number of training images used from the MinneApple training set. Standard pre-training methods are marked as dots, and our specialized pre-training is shown in blue stars. Our method achieves a better accuracy than traditional approaches, especially with only a few annotated images.



(a) Zero-shot results. (b) Two-shot results.

Fig. 6: Qualitative results for zero-shot and two-shot apple segmentation. Predicted fruit on the tree are colored cyan and predicted fruit on the ground are colored magenta. The red box highlights the unsupervised learning of the fruit on the ground class. With only two images, highly-occluded fruit are detected as shown in the yellow boxes.

samples were added to the training set.

Similar to previous work, we found that pre-training using ImageNet significantly improved the accuracy of semantic segmentation. The quantitative results in Table II and Table III show that pre-training benefits the task transfer for both few-shot and full-shot segmentation. Additionally, we confirmed that specialized pre-training on similar datasets greatly benefits few-shot fruit segmentation and it also provides a slight increase in accuracy for full-shot segmentation as seen in Table I and Fig. 5. We can also see that pre-training on CitDet allows for accurate zero-shot segmentation of in-orchard apples and it even outperforms the two-shot model with a traditional ImageNet transfer. We note that the two-shot CitDet model outperforms the four-shot model. This indicates that the model can transfer to the task of fruit segmentation with only a few examples, but it requires a larger number of samples to learn the difference in annotation styles between the pre-training and target datasets. The results of the ablation study in Table IV further shows that the addition of the ADB-BAG is responsible for the FSL of the fruit segmentation task.

Interestingly, the predictions of the models pre-trained on CitDet for the fruit on the ground class improved when fine-tuning on a small number of labeled images despite only having labels for fruit on the trees. This demonstrates that models can adapt the knowledge learned from a pre-training dataset in an unsupervised manner to the target dataset. The unsupervised learning can be seen in the qualitative results of Fig. 6. We hypothesize that this unsupervised learning is able to occur due to the similarity between the two fruit classes. As seen in the qualitative visualizations, models first learn to predict all the fruit as fruit on the tree, indicating that the fruit on the ground class is closely related. Since the CitDet models have already learned the distinction between fruit on the tree and fruit on the ground, the similarities between the two classes allow for unsupervised learning of one without labels. This could suggest that the two classes fall under a hierarchical fruit class in which all fruit is first identified before being further classified by location.

V. Conclusion and Future Work

In this paper we proposed a framework for few-shot fruit segmentation consisting of a three-branch decoder. The decoder was designed for detecting similar shape between fruits and leverages specialized pre-training on a dataset with a high similarity to the target. Our results indicate that specialized pre-training significantly improves few-shot semantic segmentation for in-orchard apples and gives a slight benefit over ImageNet pre-training for full-shot segmentation. Since this specialized pre-training has seen benefits in other areas, future few-shot segmentation work in all domains should consider the applicability of specialized pre-training for the intended target task. The qualitative results of unsupervised learning of the fruit on the ground class indicates potential future work for unsupervised transfer learning between classes included in both the pre-training and target datasets, but only labeled in the pre-training dataset.

Acknowledgments

This material is based upon work supported by the United States Department of Agriculture (USDA) under Agricultural Research Service CRIS Project #6066-21310-006-000-D.

REFERENCES

[1] C. Zhang, J. Valente, L. Kooistra, L. Guo, and W. Wang, "Orchard management with small unmanned aerial vehicles: A survey of sensing and analysis approaches," *Precision Agriculture*, vol. 22, no. 6, pp. 2007–2052, 2021.

[2] J. A. James, H. K. Manching, M. R. Mattia, K. D. Bowman, A. M. Hulse-Kemp, and W. J. Beksi, "Citdet: A benchmark dataset for citrus fruit detection," *arXiv preprint arXiv:2309.05645*, 2023.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[4] A. Parnami and M. Lee, "Learning from few examples: A summary of approaches to few-shot learning," *arXiv preprint arXiv:2203.04291*, 2022.

[5] N. Catalano and M. Matteucci, "Few shot semantic segmentation: A review of methodologies and open challenges," *arXiv preprint arXiv:2304.05832*, 2023.

[6] N. Häni, P. Roy, and V. Isler, "Minneapple: a benchmark dataset for apple detection and segmentation," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 852–858, 2020.

[7] https://github.com/robotic-vision-lab/Few-Shot-Fruit-Segmentation-Via-Transfer-Learning.

[8] O. Sener and S. Savarese, "Active learning for convolutional neural networks: A core-set approach," *arXiv preprint arXiv:1708.00489*, 2017.

[9] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy, "Speed/accuracy trade-offs for modern convolutional object detectors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7310–7311.

[10] S. Kornblith, J. Shlens, and Q. V. Le, "Do better imagenet models transfer better?" in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2661–2671.

[11] D. Hendrycks, K. Lee, and M. Mazeika, "Using pre-training can improve model robustness and uncertainty," in *Proceedings of the International Conference on Machine Learning*, 2019, pp. 2712–2721.

[12] M. Huh, P. Agrawal, and A. A. Efros, "What makes imagenet good for transfer learning?" *arXiv preprint arXiv:1608.08614*, 2016.

[13] G. Neuhold, T. Ollmann, S. Rota Bulo, and P. Kontschieder, "The mapillary vistas dataset for semantic understanding of street scenes," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2017, pp. 4990–4999.

[14] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3213–3223.

[15] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: Simple and efficient design for semantic segmentation with transformers," in *Proceedings of the Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 12 077–12 090.

[16] X. Li, L. Zhang, A. You, M. Yang, K. Yang, and Y. Tong, "Global aggregation then local distribution in fully convolutional networks," *arXiv preprint arXiv:1909.07229*, 2019.

[17] X. Zhang, H. Xu, H. Mo, J. Tan, C. Yang, L. Wang, and W. Ren, "Dcnas: Densely connected neural architecture search for semantic image segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 13 956–13 967.

[18] J. Zhao, J. Tow, and J. Katupitiya, "On-tree fruit recognition using texture properties and color data," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005, pp. 263–268.

[19] M. W. Hannan, T. F. Burks, and D. M. Bulanon, "A machine vision algorithm combining adaptive segmentation and shape analysis for orange fruit detection," *Agricultural Engineering International: CIGR Journal*, 2009.

[20] S. Arivazhagan, N. Shebiah, S. Nidhyanandhan, and L. Ganesan, "Fruit recognition using color and texture features," *Journal of Emerging Trends in Computing and Information Sciences*, vol. 1, no. 2, pp. 90–94, 2010.

[21] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[22] S. Bargoti and J. P. Underwood, "Image segmentation for fruit detection and yield estimation in apple orchards," *Journal of Field Robotics*, vol. 34, no. 6, pp. 1039–1060, 2017.

[23] S. W. Chen, S. S. Shivakumar, S. Dcunha, J. Das, E. Okon, C. Qu, C. J. Taylor, and V. Kumar, "Counting apples and oranges with deep learning: A data-driven approach," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 781–788, 2017.

[24] X. Liu, S. W. Chen, S. Aditya, N. Sivakumar, S. Dcunha, C. Qu, C. J. Taylor, J. Das, and V. Kumar, "Robust fruit counting: Combining deep learning, tracking, and structure from motion," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2018, pp. 1045–1052.

[25] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018.

[26] M. Rahnemoonfar and C. Sheppard, "Deep count: fruit counting based on deep simulated learning," *Sensors*, vol. 17, no. 4, p. 905, 2017.

[27] R. Barth, J. IJsselmuiden, J. Hemming, and E. J. Van Henten, "Data synthesis methods for semantic segmentation in agriculture: A capsicum annuum dataset," *Computers and Electronics in Agriculture*, vol. 144, pp. 284–296, 2018.

[28] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[29] M. Orsic, I. Kreso, P. Bevandic, and S. Segvic, "In defense of pre-trained imagenet architectures for real-time semantic segmentation of road-driving images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 607–12 616.

[30] H. Li, P. Xiong, H. Fan, and J. Sun, "Dfanet: Deep feature aggregation for real-time semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9522–9531.

[31] M. Gamal, M. Siam, and M. Abdel-Razek, "Shuffleseg: Real-time semantic segmentation network," *arXiv preprint arXiv:1803.03816*, 2018.

[32] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6848–6856.

[33] P. Hu, F. Perazzi, F. C. Heilbron, O. Wang, Z. Lin, K. Saenko, and S. Sclaroff, "Real-time semantic segmentation with fast attention," *IEEE Robotics and Automation Letters*, vol. 6, no. 1, pp. 263–270, 2020.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[35] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 325–341.

[36] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. Sang, "Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation," *International Journal of Computer Vision*, vol. 129, pp. 3051–3068, 2021.

[37] Y. Hong, H. Pan, W. Sun, and Y. Jia, "Deep dual-resolution networks for real-time and accurate semantic segmentation of road scenes," *arXiv preprint arXiv:2101.06085*, 2021.

[38] J. Xu, Z. Xiong, and S. P. Bhattacharyya, "Pidnet: A real-time semantic segmentation network inspired by pid controllers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 19 529–19 539.

[39] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 679–698, 1986.

[40] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," *arXiv preprint arXiv:2304.02643*, 2023.

[41] W. Falcon and the PyTorch Lightning Team, "PyTorch Lightning," 2019. [Online]. Available: https://github.com/Lightning-AI/lightning