A Comparison of Deep Learning Models for Cherry Tree Branch Segmentation

Benjamin R. Hillen
Department of Electrical Engineering and Computer Science
Oregon State University
Corvallis, OR 97331
hillenb@oregonstate.edu

Abstract

Automated cherry tree pruning requires vision systems capable of identifying branches and specific branch structures under variable weather and lighting conditions. We present a comparison of recent image segmentation models applied to modern cherry tree orchards. We evaluate the performance of these models on the task of identifying branches and branch classes in images of a trellis-orchard. We also deliver an API that enables users to create, train, and compare these models on new data. These models are intended to be applied to future work involving automated pruning systems by providing information about branches in a given image which can be used to identify pruning points on the tree.

1 Introduction

Dormant-season pruning is one of the most demanding activities associated with fruit orchards, as shown by Schouterden et al. (8). Autonomous pruning presents a possible solution that could greatly mitigate the strain placed on orchard workers and improve the production of orchards that lack sufficient numbers of workers. Therefore, there is a growing need for autonomous workers that can function in modern fruit orchard environments.

The development of accurate and reliable vision systems is critical to the success of autonomous pruning agents. An agent must be able to identify the overall structure of a tree to be able to determine if the tree requires pruning and precisely where to prune. Modern canopy and trellis orchards are noisy environments with variable weather and lighting conditions that can obfuscate attempts to segment out branches and branch classes in an image. Some work has been done on canopy-style orchards under controlled lighting and weather conditions using networks and architectures from the mid to late 2010s. However, there have been recent developments in state-of-the-art image segmentation networks that offer universal architectures and leverage transformer architectures to achieve better performance on the standard COCO datasets.

In this paper, we present a review of current deep learning models and evaluate their efficacy in a typical noisy orchard environment. We test each model using images at different times of day under varying lighting conditions in a cherry tree trellis-based orchard to evaluate the robustness of each model. We present the dataset used for training and testing each model and the COCO metrics that will be used to compare the performance of each model. The repository for this project is publicly available at `https://github.com/Ben-Hi/MS_Capstone.git`.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

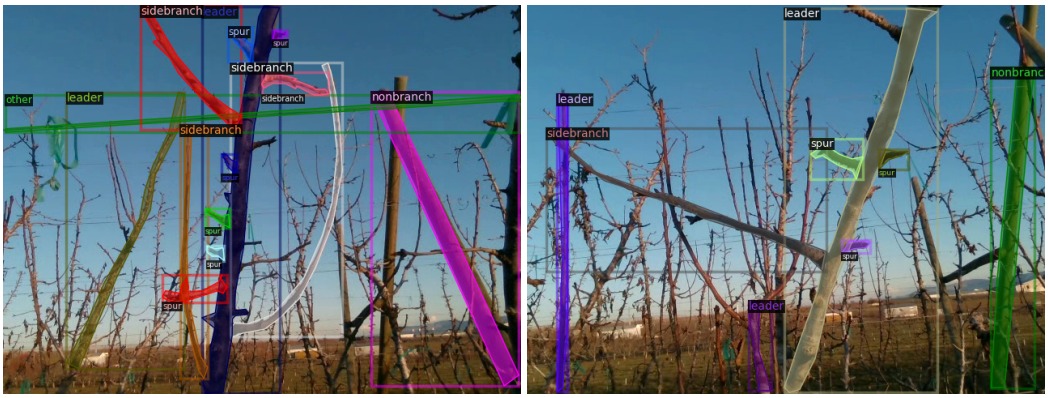


Figure 1: Sample annotated images from the test dataset. Annotations for training and testing are stored in the Labelme format and loaded into the Detectron2 standard format during training and evaluation.

2 Background and Related Work

The work done by Amatya et al. (1) cites the sweet cherry production (over half of the total amount of sweet cherries harvested in the US) in Washington State to provide an incentive to pursue automated workers in a fruit orchard environment. These reasons include high monetary and labor costs incurred by harvesting, as well as the decline in laborers. This paper focused on the harvesting stage of fruit orchard production and used a Bayesian classifier to do pixel-based segmentation of cherry branches in a canopy orchard in the harvesting season. All data used was taken at night.

The work done by You et al. (11) presents a complete autonomous pruning agent with a vision pipeline for branch segmentation. The first step in the pipeline implements the system presented in You et al. (10), which uses a generative adversarial network (GAN) and the Seg2Flow optical flow network made by Cheng et al. (4) to produce 2-channel foreground masks. This isolates the foreground branches of interest and the agent's pruning cutters. The foreground masks are then used to train a Mask R-CNN model to perform instance segmentation to detect categories of branches and the cutters used by the agent. The raw RGB data used to train the GAN was collected from a trellis orchard at varying times of day and varying weather conditions.

The work by Amatya et al. (1) uses Bayesian classifiers, while You et al. (11) relies on pre-processing images through a GAN to produce foreground masks before conducting instance-based segmentation using Mask R-CNN. While Mask R-CNN is more recent and higher-performing than Bayesian classifiers, we believe that the recent benchmark data from the Mask2Former transformer-based network by Cheng et al. (3) produces superior results compared to Mask R-CNN. Additionally, Mask2Former offers the functionality of a universal model that can perform semantic and panoptic segmentation in addition to instance segmentation. This functionality would allow for more flexibility in applications of a vision-based AI than one that uses Mask R-CNN, as a single Mask R-CNN model is only capable of performing one of these segmentation tasks.

3 Technical Approach and Methods

The vision pipeline presented by You et al. (11) relied on training two networks: a GAN to produce foreground masks for branches and a Mask R-CNN model to perform instance segmentation. Instead of training segmentation networks with foreground masks, we directly train on the RGB images in the dataset. This eliminates the use of an intermediary GAN and simplifies the comparison process between models. We will present the dataset used in this project first, then we will present the architectures of Mask R-CNN and Mask2Former.

3.1 Data

The data used for training and evaluating models in this project consists of 386 annotated RGB images from a trellis cherry orchard during the dormant season at different times of day with mostly

Table 1: Training data class balance

Class	Num in train	% of classes in train
leader	628	14.0%
sidebranch	872	19.4%
nonbranch	617	13.7%
other	1348	30.0%
spur	1028	22.9%

Table 2: Testing data class balance

Class	Num in test	% of classes in test
leader	165	15.1%
sidebranch	298	27.2%
nonbranch	155	14.2%
other	143	13.1%
spur	333	30.4%

clear weather conditions. This dataset is taken from the dataset used in You et al. (11). Each image is sized as 640x480 in png format. There are 5 classes in the dataset that correspond exactly to the classes in You et al. (11), which are:

- "leader": the primary branch of a tree in a trellis orchard. Fruiting branches grow off of this main branch.
- "sidebranch": branches that grow off of a leader branch. Produces fruit until a certain length. All wood after this length no longer produces fruit and should be pruned to make room for new fruiting wood.
- "nonbranch": objects in the foreground that are not branches, such as trellis wires.
- "other": branches that are not part of the focus tree but are in the foreground near the edge of the image.
- "spur": small protruding branches that will grow into fruiting sidebranches. Spurs should NOT be pruned.

The data was split into a training set of 308 images (80%) and a testing set of 78 images (20%). See Fig. 1 for sample images. See Tables 1 and 2 for analyses of the class distributions.

3.2 Mask R-CNN

The Mask R-CNN network is based on the architecture described in He et al. (6). The model architectures are chosen from the model zoo presented by Wu et al. (9) that implements standard networks that are pre-trained on the 2017 COCO training dataset for approximately 37 COCO epochs. The Mask R-CNN models used in this project utilize a residual network (ResNet) backbone with a feature pyramid network (FPN). The ResNet architecture is discussed in He et al. (5).

The first Mask R-CNN model used in this project uses ResNet-50 as its backbone. This model contains 48 convolutional layers: one layer for max pooling and one layer for average pooling.

The second Mask R-CNN model uses ResNet-101 as its backbone. This model has a similar structure to ResNet-50 but contains 51 additional convolutional layers.

Both Mask R-CNN models were trained for 80,000 epochs with a learning rate of 0.001 and no learning rate decay. Due to limited GPU resources, the batch size was set to 2 images per batch. Both models used Stochastic Gradient Descent with momentum as the optimizer.

3.3 Mask2Former

Mask2Former was developed in 2022 by Cheng et al. (3) to unify the panoptic, semantic, and instance sub-problems of image segmentation. The result is a universal architecture that can perform any of

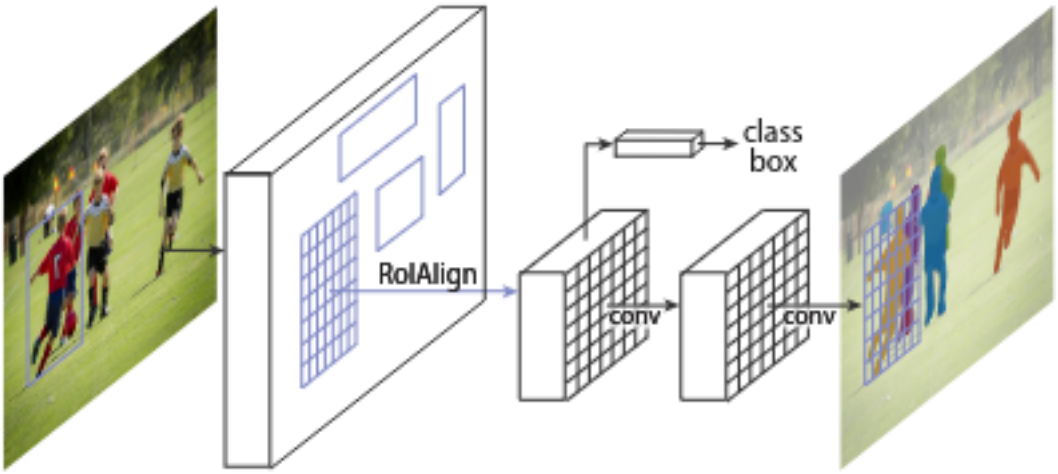


Figure 2: Mask R-CNN architecture as illustrated in He et al. (6).

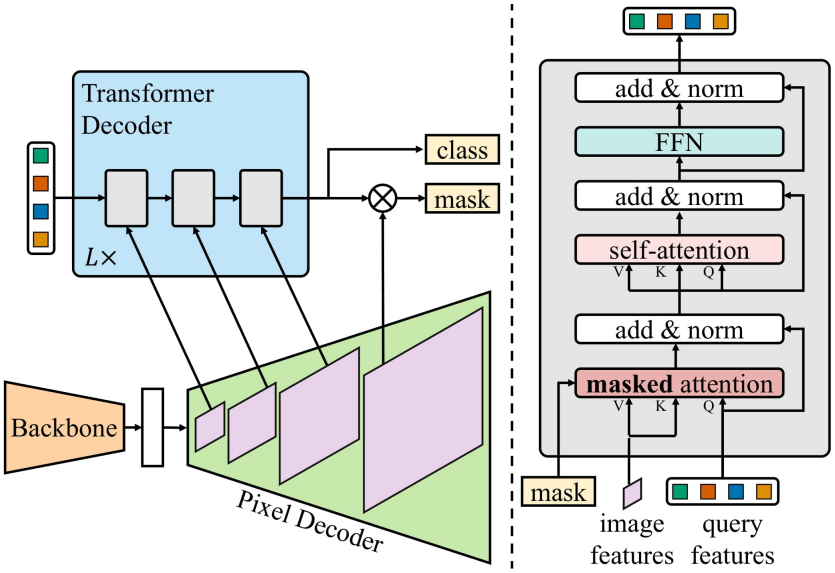
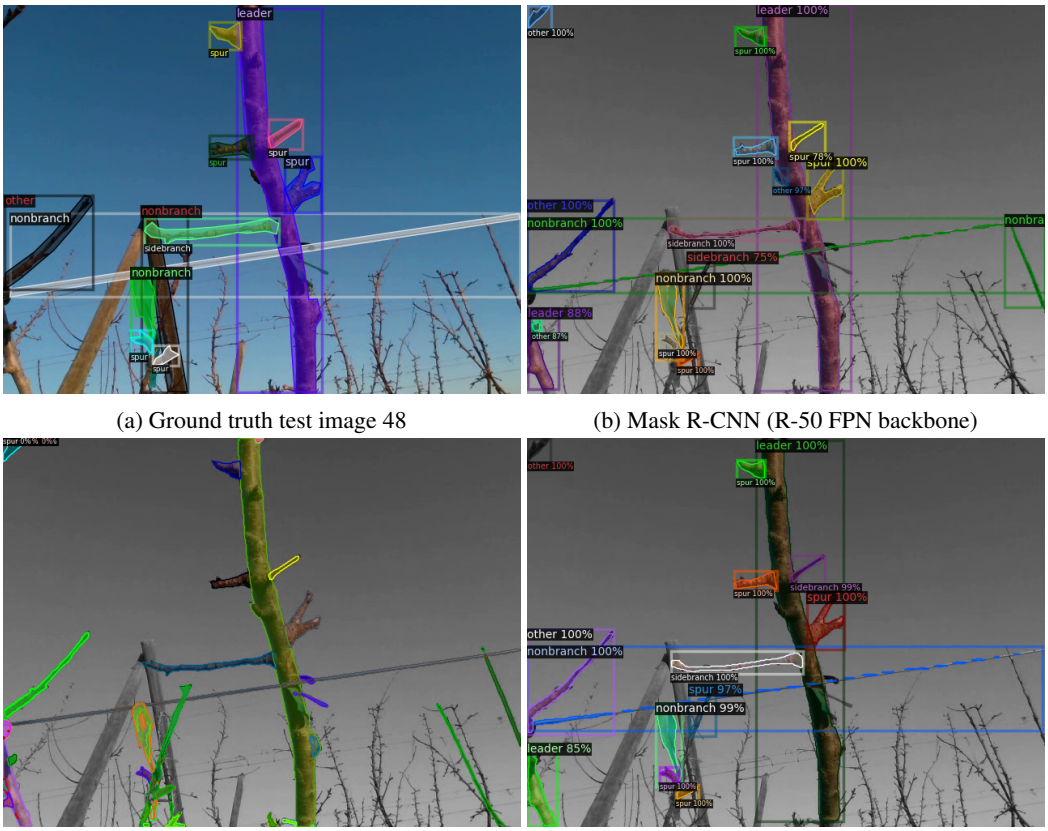


Figure 3: Mask2Former architecture as illustrated in Cheng et al. (3).

the instance segmentation sub-problems and has produced state-of-the-art results in each sub-problem. Refer to Fig. 3 for the architecture diagram presented by Cheng et al. (3).

Mask2Former extracts an image feature map using a pixel-level module and upsamples the map with pixel-level decoder, both of which are drawn from the original MaskFormer from Cheng et al. (2). The upsampled per-pixel embeddings are then used by a Transformer decoder to generate masks and categorical classifications for objects in the image. The key contribution of the Transformer decoder is the implementation of localized context. During normal cross-attention in the Transformer decoder, the attention matrix is densely populated by computing the product of the input queries and the input keys. However, Mask2Former introduces a modulation mechanic called "masked attention", where the output of the query-key product at location (x, y) is set to 0 if that feature is not included in the mask at that layer of the decoder. This forces the computation of attention to be focused on local features instead of the whole feature map as in traditional transformer-based segmentation models.

In this project, we use a tiny Swin (Swin-t) backbone to produce the feature maps for Mask2Former. The smaller Swin backbone was chosen for the best trade-off between baseline precision as presented by Cheng et al. (3) and the GPU memory constraints of the machine that the project was conducted



(a) Ground truth test image 48

(b) Mask R-CNN (R-50 FPN backbone)

(c) Mask2Former (Swin-t backbone)

(d) Mask R-CNN (R-101 FPN backbone)

Figure 4: Sample comparison between ground truth masks (top left) and predicted masks for all models on one image from the test set. For clarity, the non-annotated areas of the predicted image have been greyed out. Mask2Former does not predict bounding boxes due to its architecture.

on. Mask2Former used the ADAM optimizer to provide faster training times at the tradeoff of less optimal convergence compared to SGD. This was necessary due to hardware limitations.

4 Results

All models presented in this section were trained for 80,000 epochs and evaluated using the COCO evaluation metrics described below.

Each model is evaluated using the COCO object detection evaluation metrics. The main metrics used are the average precision (AP) and average recall (AR) of the model. COCO evaluation abbreviates mean average precision (mAP) to AP and leaves the interpretation of which metric is used up to the reader. Typically, mAP is the average across all categories. This report maintains the COCO convention of using AP to mean mAP. We also provide a table of per-class AP values for further analysis of model efficacy.

In the calculation of COCO AP and AR, the values of each metric are averaged over 10 IoU thresholds starting at IoU = 0.5 and ending at IoU = 0.95 with a step-size of 0.05, which produces a weighted value depending on the strictness of the IoU threshold. Calculating over a range of IoU values produces evaluation metrics that prefer models with higher levels of localization. For more information on the full set of COCO object detection metrics, see `https://cocodataset.org/#detection-eval`.

Overall, we find that Mask2Former is able to outperform both Mask R-CNN models in the primary AP metric. Additionally, we find that Mask2Former is more flexible in terms of future applications to the problem of image segmentation in noisy orchard environments.

Table 3: Per-class segmentation AP for Mask R-CNN ResNet-50 and ResNet-101 alongside Mask2Former Swin-t with an IoU threshold range of 0.5:0.95 for AP calculation.

	Segmentation AP		
Class	R50	R101	Swin-t
leader	38.9%	40.4%	45.7%
sidebranch	13.9%	13.6%	18.7%
nonbranch	2.7%	2.9%	4.5%
other	1.2%	1.4%	2.3%
spur	11.5%	10.0%	11.4%

Table 4: AP and AR for Mask R-CNN R-50 and R-101 alongside Mask2Former Swin-t with an IoU threshold range of 0.5:0.95 used for AP calculation.

	R50		R101		Swin-t	
Prediction Type	AP	AR	AP	AR	AP	AR
segmentation	13.6%	22.4%	13.6%	23.5%	16.5%	28.1%
bounding box	29.2%	41.4%	29.6%	42.2%	N/A	N/A

When comparing Mask R-CNN networks to each other, we find that the APs of both R-50 and R-101 are roughly equal in both per-class segmentation in Table 3 and total AP in Table 4. Furthermore, there is a notable difference in AP and AR for both R-50 and R-101, indicating that both models had fairly high false-positive rates across all categories.

When comparing Mask R-CNN networks to the Mask2Former Swin-t network, we find that Mask2Former outperforms both R-50 and R-101 in terms of total segmentation AP and all per-class AP values, with the exception of the spur class against R-50. Again, we note a similar difference in AP and AR in Swin-t. Some possible solutions that could address this issue include performing hyperparameter tuning for all models (the models in this project were untuned from their pre-trained starting networks) and the use of augmentation on the training dataset. Given that this project is performed in the context of identifying cherry tree architecture for automated pruning of dead sidebranch material, we find that the higher per-class AP of Swin-t on the "leader" and "sidebranch" classes supports our claim that Mask2Former Swin-t is better suited to the task of image segmentation in a noisy orchard environment than Mask R-CNN.

Note that the "nonbranch" and "other" classes are the most difficult to segment for all models. As shown in Table 1 and Table 2, the proportion of the "other" class in the testing set is 56% smaller than the proportion of the "other" class in the training set. This "other" class also has a high likelihood of confounding the "nonbranch" class. The "other" class is used to identify branches that are not focused in the image while still being in the foreground, according to You et al. (11), while the "nonbranch" class identifies objects that are not branches in the foreground, such as trellis wires and materials. These objects are typically less prominent in terms of size on the image compared to typical "leader" objects, which usually occupy a proportionally large amount of pixels compared to single instances of the other classes.

The potential confounding quality of the "nonbranch" class is reflected qualitatively in Fig 4. The "nonbranch" wire that runs through the center of the image, the wooden trellis material in the bottom left, and the small objects appearing on the left and right sides of the image all present difficult masking challenges. R-50 and R-101 are able to draw accurate bounding boxes around the right-side nonbranch, but the mask is not fully connected and R-50 incorrectly identifies a background branch as a nonbranch. Mask2Former also incorrectly identifies the background branch but is able to produce a fully connected mask of the trellis wire. Note that all networks are able to clearly identify the single leader branch, the sidebranches, and the spurs in the image in comparison to attempts to identify nonbranch objects. This confounding effect appears to be one of the biggest challenges for any attempt at segmentation in the trellis environment.

5 Summary and Future Direction

We presented a comparison between Mask R-CNN and Mask2Former on the task of image segmentation in a noisy orchard environment and described the history of image segmentation in orchards. The dataset that was provided for this project was presented, along with an overview of the balance of the dataset. The architectures of Mask R-CNN and Mask2Former were described and referenced. We find that under the COCO evaluation metrics, Mask2Former outperforms Mask R-CNN in the task of instance segmentation of cherry tree branches. Because of the differences in architecture and design philosophies of the networks, we find that Mask2Former allows for more flexibility in future applications of orchard image segmentation due to its ability to provide instance, semantic, and panoptic segmentation in one model, while Mask R-CNN requires separate models to implement all of these tasks. We also identified the confounding nature of the "nonbranch" class in the dataset and how the presence of nonbranch objects in a trellis environment poses a challenge to image segmentation.

For future work on the project, we recommend that the OneFormer model by Jain et al. (7) be compared against Mask2Former on the task of instance segmentation in the cherry orchard dataset used by this project. OneFormer has the potential to be trained only once on a dataset and may then be used across all forms of image segmentation, such as panoptic, semantic, and instance segmentation. While Mask2Former provides support for all image segmentation masks, the single model must be trained individually on separate datasets for each desired task. OneFormer would offer the advantage of only needing to train once to provide state-of-the-art results on any segmentation task that could be needed in an orchard environment.

Additionally, we would also recommend longer training sessions for all models if resources permit. The models in this project were trained on a single NVIDIA GeForce 3060 for 80,000 iterations per model. This also limited the size of models that could be trained for the project, so the use of Swin-base and Swin-large backbones for Mask2Former was not possible and should be considered for future comparisons. If a specific model is chosen for use in an automated orchard environment, we would recommend the use of RayTune for hyperparameter optimization to further improve the model's performance. Analyzing the effects of different dataset sampling methods could also clarify some of the shortcomings of the models in this paper in terms of low AP on specific classes that are either underrepresented in the testing vs. training split or possibly introducing a confounding element to other classes.

References

[1] Suraj Amatya, Manoj Karkee, Aleana Gongal, Qin Zhang, and Matthew D. Whiting. Detection of cherry tree branches with full foliage in planar architecture for automated sweet-cherry harvesting. *Biosystems Engineering*, 146:3–15, 2016. ISSN 1537-5110. doi: https://doi.org/10.1016/j.biosystemseng.2015.10.003. URL `https://www.sciencedirect.com/science/article/pii/S1537511015001683`. Special Issue: Advances in Robotic Agriculture for Crops.

[2] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation, 2021.

[3] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation, 2022.

[4] Jingchun Cheng, Yi-Hsuan Tsai, Shengjin Wang, and Ming-Hsuan Yang. Segflow: Joint learning for video object segmentation and optical flow, 2017.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[6] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn, 2018.

[7] Jitesh Jain, Jiachen Li, MangTik Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi. Oneformer: One transformer to rule universal image segmentation, 2022.

[8] Gert Schouterden, Rafaël Verbiest, Eric Demeester, and Karel Kellens. Robotic cultivation of pome fruit: A benchmark study of manipulation tools—from research to industrial standards. *Agronomy*, 11(10), 2021. ISSN 2073-4395. doi: 10.3390/agronomy11101922. URL https://www.mdpi.com/2073-4395/11/10/1922.

[9] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

[10] Alexander You, Cindy Grimm, and Joseph R. Davidson. Optical flow-based branch segmentation for complex orchard environments, 2022.

[11] Alexander You, Nidhi Parayil, Josyula Gopala Krishna, Uddhav Bhattarai, Ranjan Sapkota, Dawood Ahmed, Matthew Whiting, Manoj Karkee, Cindy M. Grimm, and Joseph R. Davidson. An autonomous robot for pruning modern, planar fruit trees. *JOURNAL OF ROBOTICS AND AUTOMATION LETTERS*, 1, 6 2022. doi: 10.48550/arxiv.2206.07201. URL https://arxiv.org/abs/2206.07201v1.