Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Automatic Blossom Detection in Apple Trees using Deep Learning

Uddhav Bhattarai*, Santosh Bhusal*, Yaqoob Majeed*, Manoj Karkee*

**Center for Precision and Automated Agricultural Systems, Department of Biological Systems Engineering, Washington State University, Pullman, WA USA, Corresponding author (e-mail: uddhav.bhattarai@wsu.edu)*

Abstract: Overcropping in fruit trees results in decreased fruit size, poor fruit quality, biennial bearing, and reduction in productive life of orchards. Although flowers and fruits are removed/thinned naturally, they require additional thinning for commercial grade fruit production. Integration of machine vision system in mechanical/chemical thinning facilitates automated selective blossom thinning. The primary requirement for automating blossom thinning is to estimate the blossom density in apple trees under varying background and lighting conditions. In this work, we implement Mask-RCNN algorithm to perform instance segmentation of apple blossoms. Different image augmentation techniques were implemented and their impact on blossom detection were assessed. Experiments were conducted to achieve optimal values of hyperparameters of the deep learning network during the training. Implementation of image augmentation was crucial to reduce validation loss and improve detection accuracy of segmentation algorithm. The proposed system achieved average precision (AP) of 0.86 in detecting blossoms in test dataset previously unseen by the network.

Keywords: Pattern recognition and Artificial Intelligence in agriculture, Agricultural robotics, Precision farming, Blossom detection, Blossom intensity estimation, Instance segmentation, Blossom thinning

1 INTRODUCTION

Fruit trees often bloom more flowers/blossoms and set more fruits than the desired amount for achieving target yield, size, and quality in commercial farming. Unlike stone fruits, pome fruits like apples produce clusters of flowers and fruits in each bud. Crop thinning, therefore, is crucial as it discourages overbearing and early fruit drop, improves fruit size, color and overall quality, reduces limb damage and avoid biennial bearing. Current blossom thinning approaches involve hand thinning, chemical thinning, and mechanical thinning (Bound, 2018; Wang et al., 2013). Hand thinning involves removing excess blossoms manually. Although, hand thinning is an effective method of removing blossom, growers are facing difficulty in finding sufficient workers and accommodating the increasing cost of farm labors (Glozer and Hasey, 2006; Hertz and Zahniser, 2013). On the other hand, chemical and mechanical thinning suffer from high variability and uncertainly in thinning results due to variation in environment, weather parameters, canopy density, canopy types and density of blossom and green fruit.

Robotic thinning systems require a robust sensing system in addition to precision thinning devices to identify, locate and effectively remove unwanted flowers. One of the major challenges in vision-based blossom thinning is to develop robust and accurate blossom detection algorithms that can detect apple blossoms in varying background, noise, and lighting conditions typical in an orchard environment. There have been a few research efforts in developing machine vision systems for blossom detection and localization. The detection algorithms are based on extraction of feasible color channel followed by contrast variation (Gebbers et al., 2013), threshold operation, and morphological image processing (Krikeb et al., 2017; Xiao et al., 2014; Aggelopoulou et al., 2011; Hočevar and Demšar, 2014). However, these color thresholding and morphological operations have limited capability to minimize the effect of varying lighting condition, varying shape and texture, and random noise present in the field environment. Recently, there are promising implementations of machine learning and deep learning techniques in blossom and green fruit detection in outdoor environment (Dias et al., 2018a; Dias et al., 2018b; Bargoti and Underwood, 2017). Researchers have used FasterRCNN for fruit detection in apples, mangoes, almonds as well as flower detection in apples (Bargoti and Underwood, 2017; Farjon et al., 2019) Considering high blossom density in full bloom period, bounding box based Faster RCNN is not capable to provide desired precise pixel level blossom segmentation. Dias et al. (2018a) implemented semantic segmentation of blossom through a pipeline involving iterative computation of region proposals followed by feature extraction using CNN, dimensionality reduction by PCA, and classification using SVM. While this approach performed better compared to conventional machine learning techniques, the computation of region proposal via super pixel is prone to image variations, which might not lead to optimal region proposals. As the complete

detection architecture is based on the segmentation of region proposals, the generalizability of this approach is likely to be inferior than end-to-end architecture (Jiang and Li, 2020; Dias, 2018b). Improving their previous work, Dias et al. (2018b) proposed division of each image into grids, CNN application for foreground and background segmentation in each grid followed by refine of region growing approach. The system outperformed Dias et al. (2018a) in terms of accuracy. However, the system has room for improvement in evaluation time as it took 50 seconds, on average for each image. Furthermore, creating portraits, storing, and loading prediction scores adds computational overhead.

Deep learning techniques have been proven to be more robust, accurate, and reliable in various computer vision-based object detection methods. Transfer learning takes advantage of existing knowledge of related task or domain to increase the learning speed and performance of the system by fine-tuning the pretrained models using corresponding dataset (Kamilaris and Prenafeta-Boldu, 2018). The generalizability of the detection algorithm in multiple environments, their applicability in real time vision-based robotics are two major challenges of implementing neural networks in industry. Further research and development are necessary to make these systems robust and widely applicable to the targeted blossom detection and thinning.

Our approach utilizes a unified end-to-end instance segmentation architecture that takes a single image as an input and returns the all instances of flowers (blossom detection + classification at pixel level) without any pre-processing. We report average detection speed of 1.27 seconds for images with a pixel resolution of 1920 x 1080. Mask Region-based Convolution Neural Network (Mask R-CNN) is extension of Faster R-CNN (He et al., 2017; Ren et al., 2015), which can perform pixel level segmentation and mask generation for the objects in images. An existing Mask R-CNN algorithm proposed originally by Facebook AI Research (FAIR) was fine-tuned to detect apple blossoms. This article provides the details of experimental methods, as well as performance evaluation against an image dataset acquired in a commercial apple orchard without pre-processing and background manipulation. Section 2 describes the data collection strategy, dataset preparation, and the techniques followed to implement the deep learning algorithm. Section 3 incorporates the results and discussion. Section 4 includes the conclusions derived based on the results from this work and presents a potential direction for further research and development in this area.

2 MATERIALS AND METHODS

2.1 Data Collection and Vision Sensor

The experimental data was collected in commercial apple orchards during hand blossom thinning in April 2018 and 2019 (Washington, USA). Orchards were formally trained into 2D fruiting wall architectures - vertical for Scifresh variety and V-trellised system for Envy. Data collection was performed in daylight condition without any background manipulation. Imaging sensors were positioned ~1.5m from the canopy trunk centre and ~1m above the ground reference.

This study used a low-cost Microsoft Kinect V2 time of flight based RGB-D sensor for image acquisition for both years. The sensor system constituted two cameras, namely IR camera (512 x 424) and RGB camera (1920 x 1080) with a field of view of 70 and 60 degrees, respectively (Amon et al., 2014). 2D RGB images were used for instance segmentation of apple blossoms. In future, results from blossom segmentation will be combined with the depth information which allows estimation of spatial distribution of blossoms (blossom density estimation) and development for thinning rules for vision based-thinning system.

2.2 Blossom Detection

2.2.1 Mask R-CNN

Deep learning is the extension of classical machine learning techniques for hierarchical feature learning. The main advantage of deep learning is its capability to create and learn from features at different levels (higher-level features can be formed using the lower level features) (Kamilaris and Prenafeta-Boldu, 2018). Mask R-CNN is the extension of Faster R-CNN for predicting segmentation masks on each Region of Interest (RoI) (He et al., 2017; Ren et al., 2015), see Fig. 1. Mask R-CNN employs identical first stage computation of RPN as Faster R-CNN while modifying the second stage with additional branch for segmentation mask computation. (He et al., 2017).

2.2.2 Network Implementation

In this work, the implementation of Mask R-CNN was based on Feature Pyramid Network (FPN) with ResNet101 as backbone (Lin et al., 2017; He et al., 2016). ResNet uses residual learning mechanism that reduces number of

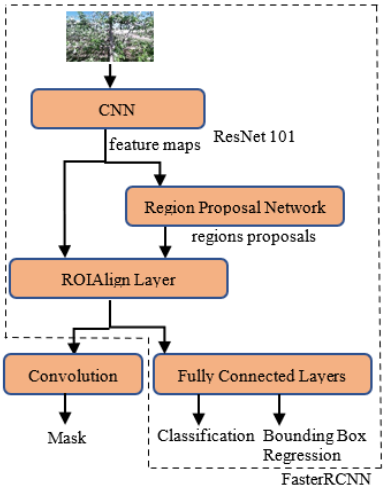

Fig. 1: Architecture of Mask RCNN based instance segmentation framework

parameters to be tuned and hence reduces overall computational cost. Furthermore, residual links speed up the convergence of deep learning network (Khan et al., 2019). The Mask R-CNN implementation in this work was extended from the existing implementation from Matterport Inc. (Sunnyvale, CA) released under MIT License (Abdulla, 2017). The system is Keras implementation with TensorFlow as the backend engine for low-level computations. We employed transfer learning by using pretrained model trained in MSCOCO image dataset (Lin et al., 2014). To determine the optimal number of layers to train, the ResNet101 backbone was trained up to 100 epochs using three different strategies *i)* training all network layers with randomized initial weights, *ii)* training convolution layer five and up, *iii)* training the RPN, classifier, and mask heads of the network. The network was trained using stochastic gradient descent with momentum of 0.9 and learning rate of 0.0005. Low value of learning rate results optimal set of weights at the cost of longer training time.

2.2.3 Image Dataset

The image dataset constituted 205 images of 1920 x 1080-pixel resolution. The entire dataset was labelled with Labelbox annotator without any pre-processing or background manipulations (Labelbox Inc.). Each image was annotated at pixel level with multiple polygonal masks indicating cluster of apple blossoms. As summarized in Table 1, 177 images collected from Scifresh and Envy orchard in 2019 (8317 flower clusters) were randomly divided into training (150), and validation (27) images. In addition, 28 Scifresh and Envy images with 1374 blossom instances collected in 2018 were used for testing the system accuracy. Since flowering location, biological properties of plant, image acquisition time, and flowering intensity varies every year, using same orchard block for acquiring training, validation, and testing images does not provide replicated result in test set.

Table 1: Division of training, testing, and validation dataset

	Orchard Blocks (Year)	# Images	# Blossom Instances
Training	Scifresh & Envy (2019)	150	6879
Validation	Scifresh & Envy (2019)	27	1438
Testing	Scifresh & Envy (2018)	28	1374
Total		**205**	**9691**

2.2.4 Data Augmentation

Data augmentation is crucial in improving overall learning and performance by enlarging the dataset artificially without taking new images. This is particularly important when available dataset is small (Kamilaris and Prenafeta-Boldu, 2018). We employed each data augmentation approach to 60% of training images which increased the training dataset from 150 images to 240 images for all augmentation techniques. To improve and assess the variability in the dataset, six different data augmentation techniques were implemented, namely: (a) Flip input images horizontally (Flip L-R), (b)Flip input images vertically (Flip U-D), (c)Rotate input images in a range of -60 to +60 degrees (Rotate), (d) Scale input images by a factor of 0.5 to 1.5 (Scale), (e) Combination of horizontal and vertical flips, rotation, and scaling in random order (Combined), (f) Comprehensive augmentation involving horizontal flip, vertical flip, random crop, gaussian blur (S.D. = 0 to 0.5), contrast variation, additive gaussian noise(mean=0, S.D.= 0 to 12.75), scaling, and rotation in random order (Comp).

2.2.5 Performance Assessment

For each of the input images, the instance segmentation algorithm outputs binary mask indicating whether the region of interest (pixel) belongs to "blossom" or "background" class. The manually labelled blossom instances were assigned as ground truth, which were compared against the detection results achieved by the proposed algorithm. Three performance matrices were observed: Precision, Recall, and Average Precision (AP).

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{1}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{2}$$

The AP presents the precision recall curve as the weighted mean of precisions at each class score threshold with the increase in recall from previous threshold used as weight. The computation of AP does not use interpolated variant of precision values. (Davis and Goadrich, 2006; Pedregosa et al., 2011). The average precision formulated in Pedregosa et al. (2011) is given as:

$$AP = \sum_t (R_t - R_{t-1}) P_t \tag{3}$$

Where, R_t and P_t are the precision and recall values for the classifier threshold t.The overall validation loss was computed as the combination of classification loss (L_{class}), bounding box loss (L_{bbox}), and the mask loss (L_{mask}) (Abdulla, 2017).

$$L_{overall} = L_{class} + L_{bbox} + L_{mask} \tag{4}$$

The class loss considers the confidence of the model in predicting the correct class. Bounding box loss takes into account of the distance between the true bounding box parameters (origin, width, height) and the predicted bounding box. Finally, the mask loss considers model confidence in the binary classification of each pixel in "blossom" or "background". Mask loss is the binary cross entropy for the pixel classification. The lower the value of the validation loss, the better the model trained with the given dataset. Please refer (Abdulla, 2017) for formulation details.

3 RESULTS AND DISCUSSION

As summarized in Table 1 a total of 27 images were used for validation while 28 images for accuracy evaluation of the implemented algorithm.

3.1 Training ResNet101 Layers

As shown in Fig. 2, among the three training methods (see section 2.2.2), training all the backbone layers surpassed performance in terms of validation loss throughout the whole training and validation process. When only network heads were trained, the network suffered from high validation loss. The performance of the network was comparatively better when training was performed for the layers five and up. Training the lower layers leveraged to extract low level key information helpful for identifying apple blossom geometry. Furthermore, up to around the eighth epoch, the validation loss for all trained layers decreased significantly with continued reduction up to the final epochs.

3.2 Image Augmentations

In the second experiment, ResNet101 backbone was trained in all layers up to 100 epochs by varying the augmentation techniques while keeping all the hyperparameters the same. The image augmentation is crucial in increasing the variability and preventing the model from overfitting. The same test dataset used for assessing the performance of training different network layers was used for assessing the detection performance under varying augmentations. Based on the precision recall metrics, the precision recall curve was plotted along with the computation of mean average precision over recall range. Fig. 3 shows that the model without any augmentation started to overfit after ~20 epochs, and the validation loss increased linearly with increase in training epochs. Among the single augmentation techniques, rotation provided lower validation loss compared to scaling,

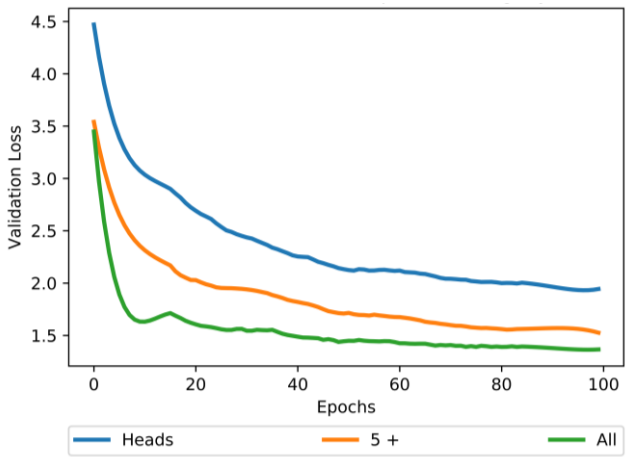

Fig. 2: Assessing training performance for three different training approaches using "combined" augmentation technique. As shown by the green curve, the validation loss computed from the model trained with all layers of ResNet-101 achieved the lowest loss among the three different methods.

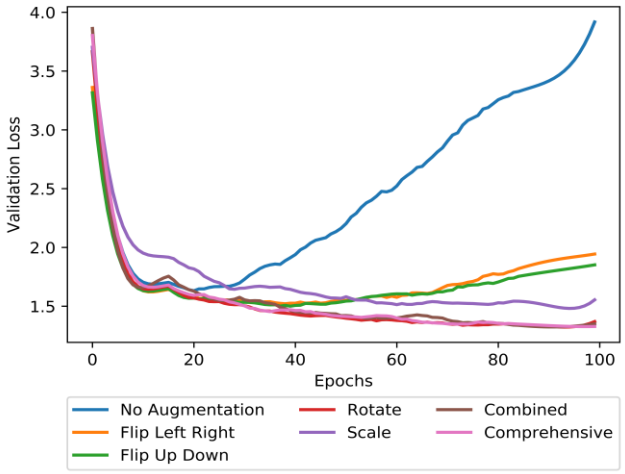

Fig. 3: Validation loss for models with various data augmentation techniques. Figure shows that the model without augmentation starts to overfit very early in the training process and the model with rotation, combined, and comprehensive augmentation have the least loss at the final epochs.

horizontal and vertical flip of image during complete training interval. As the distribution of flowers in the trees are randomly orientated in all possible directions there is no significant difference in the flip up-down and flip left-right augmentation, see Fig. 2. Relatively lower validation loss was observed with rotation, combined, and comprehensive augmentation, which is desirable to achieve enhanced object detection. However, the numerical difference of validation loss is not high at each epoch compared to flip and scaling type augmentation. The validation loss for rotation and combined augmentation seems to be increasing towards the end of the training epochs. Training can be extended to a greater number of epochs for identifying if the model starts to overfit after 97th epochs. Fig. 4 shows algorithm performance by comparing the human labelled ground truth (blue polygons) with detection results (red polygons) achieved by Mask R-CNN algorithm. The algorithm was successful in detecting majority of blossoms with some inaccuracy. Further training and fine tuning with additional dataset might help improve detection performance. Moreover, the algorithm was able to detect true blossoms that are not labelled by humans as blossoms, see Fig. 4.

The accuracy assessment in test data set did not entirely, replicate the performance trend as seen in validation loss computation, see Fig. 5. Deviating from validation loss assessment, the comprehensive and rotation augmentation performed poorly in object detection with mean average precision of 0.60 and 0.75, Table 1. It was found that the comprehensive augmentation performed much worse compared to the model without image augmentation AP=0.75. This might be the case of model overfitting where the CNN performed well in training and validation images but not in the test images. Further training and evaluation in additional dataset would be necessary to validate these results.

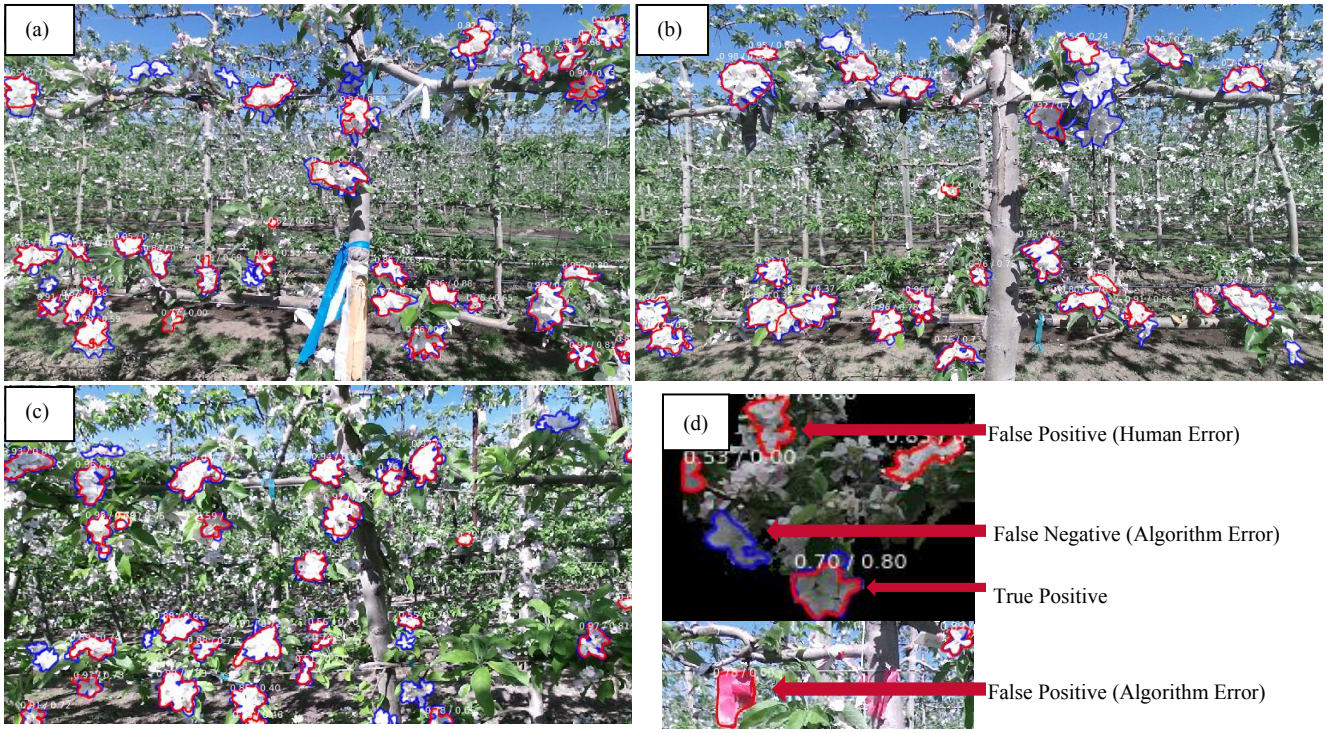

Fig. 4: Detection result achieved by Mask R-CNN algorithm compared with ground truth dataset [(a), (b): Scifresh and (c): Envy]. Objects inside blue, and red polygons indicate ground truth and detection results, respectively. (d) Representative examples of different detection scenarios

The horizontal and vertical flip tentatively displayed similar behaviour in object detection as in the validation loss computation. The horizontal and vertical flip added little variability in the data set with trivial performance improvement compared to the model without augmentation. Among all single augmentation techniques, the image scaling technique improved the overall detection with minimal additional computation overhead (AP=0.86). We implemented image scaling in a range of 0.5 to 1.5, which increased the variability in the data set without loss of information. However, when scaling was combined with the rotation, horizontal and vertical flip, that did not contribute any further improvement in AP.

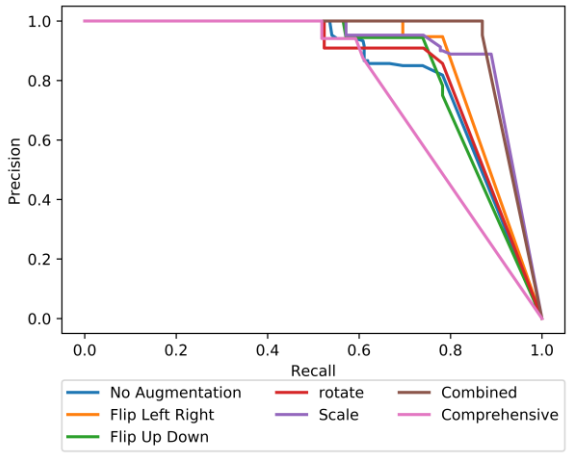

Fig. 5: Precision recall curve obtained by testing each of the models over test set. With higher area under the curve, the scaling and combined model outperformed all other models.

Table 2: AP for Intersection over Union (IOU)>50 computed from all augmentation methods over the test dataset.

	No Aug	Flip L-R	Flip U-D	Rotate	Scale	Combined	Comp
AP	0.75	0.79	0.78	0.75	0.86	0.86	0.60

The system was deployed in 24 GB NVIDA TITAN Xp, and achieved average detection speed of 1.27 seconds per image. Farjon et al. (2019) implemented Faster-RCNN to detect blossoms and reported AP = 0.68 (IOU>0.3). Our approach surpassed algorithm performance reported by Farjon et al. (2019) with AP = 0.86 (IOU>0.5).

4 CONCLUSION

Integration of machine vision system in mechanical thinning facilitates automated robotic blossom thinning. In this study we proposed deep leaning-based blossom detection algorithm and evaluated performance accuracy in images acquired in commercial apple orchard. It was observed that Mask-R-CNN based deep learning algorithm was able to detect apple flower blossoms with mean average precision of 0.86. Experiments showed that the accuracy could be significantly enhanced by combining different data augmentation techniques. The performance of the model in detecting flowers could potentially be further improved by increasing the size of the dataset and/or by employing different image augmentation techniques. However, one should be cautious about impact of image augmentation as image augmentation can sometimes result in loss of information. In the future multi class object detection will be implemented to estimate and validate the canopy parameters such as branch/trunk diameter. Information about the canopy structure will be

helpful in estimating blossom density and development of thinning rules. Furthermore, high resolution images are preferred for blossom density estimation because it facilitates additional details to be visible such that single/individual flowers detection can be improved. As resolution of images acquired by Microsoft Kinect V2 are not enough to identify individual flower within the cluster, high resolution images will be investigated for blossom density estimation in the future.

REFERENCES

Abdulla, W. 2017. Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow. Matterport Inc. Available at: https://github.com/matterport/Mask_RCNN. Accessed: June 2019.

Aggelopoulou, A. D., D. Bochtis, S. Fountas, K. C. Swain, T. A. Gemtos and G. D. Nanos. 2011. Yield prediction in apple orchards based on image processing. *Precision Agriculture* 12(3): 448-456.

Amon, C., F. Fuhrmann and F. Graf. 2014. Evaluation of the spatial resolution accuracy of the face tracking system for kinect for windows v1 and v2. In *Proceedings of the 6th Congress of the Alps Adria Acoustics Association,* 16-17.

Bargoti, S. and J. Underwood. 2017. Deep fruit detection in orchards. In *2017 IEEE International Conference on Robotics and Automation (ICRA),* 3626-3633. IEEE.

Bound, S. 2018. Getting the most out of chemical thinning. Apple & Pear Australia Ltd. Available at: https://apal.org.au/getting-chemical-thinning/. Accessed: March 2019.

Davis, J. and M. Goadrich, 2006. The relationship between Precision-Recall and ROC curves. In *Proceedings of the 23rd international conference on Machine learning*, 233-240.

Dias, P. A., A. Tabb and H. Medeiros. 2018a. Apple flower detection using deep convolutional networks. *Computers in Industry,* 9917-28.

Dias, P.A., A. Tabb and H. Medeiros, 2018b. Multispecies fruit flower detection using a refined semantic segmentation network. *IEEE Robotics and Automation Letters*, *3*(4), 3003-3010.

Farjon, G., Krikeb, O., Hillel, A.B. and Alchanatis, V., 2019. Detection and counting of flowers on apple trees for better chemical thinning decisions. *Precision Agriculture*, 1-19.

Gebbers, R., M. Pflanz, A. Betz, B. Hille, J. Mattner, T. Rachow-Autrum, M. Özyurtlu, A. Schischmanow, M. Scheele and J. Schrenk. 2013. OptiThin–Implementation of precision horticulture by tree-specific mechanical thinning. *Massendatenmanagement in der Agrar-und Ernährungswirtschaft–Erhebung–Verarbeitung–Nutzung*.

Glozer, K. and J. Hasey. 2006. Mechanical thinning in cling peach. *HortScience* 41(4), 995.

He, K., G. Gkioxari, P. Dollár and R. Girshick. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision,* 2961-2969.

He, K., X. Zhang, S. Ren and J. Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition,* 770-778.

Hertz, T. and S. Zahniser. 2013. Is there a farm labor shortage? *American Journal of Agricultural Economics* 95(2), 476-481.

Hočevar, T. and J. Demšar. 2014. A combinatorial approach to graphlet counting. *Bioinformatics* 30(4): 559-565.

Jiang, Y. and C. Li, 2020. Convolutional Neural Networks for Image-Based High-Throughput Plant Phenotyping: A Review. *Plant Phenomics*, *2020*, p.4152816.

Kamilaris, A. and F. X. Prenafeta-Boldu. 2018. Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture,* 14770-90.

Khan, A., Sohail, A., Zahoora, U. and Qureshi, A.S., 2019. A survey of the recent architectures of deep convolutional neural networks. *arXiv preprint arXiv:1901.06032*.

Krikeb, O., V. Alchanatis, O. Crane and A. Naor. 2017. Evaluation of apple flowering intensity using color image processing for tree specific chemical thinning. *Advances in Animal Biosciences* 8(2), 466-470.

Labelbox Inc. Labelbox. Available at: https://labelbox.com/. Accessed: July 2019.

Lin, T., M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár and C. L. Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision,* 740-755. Springer.

Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B. and Belongie, S., 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117-2125.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss and V. Dubourg. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research* 12(Oct): 2825-2830. Available at: https://scikit-learn.org/stable/modules/model_evaluation.html#davis2006

Ren, S., K. He, R. Girshick and J. Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems,* 91-99.

Wang, M., H. Wang, Q. Zhang, K. M. Lewis and P. A. Scharf. 2013. A hand-held mechanical blossom thinning device for fruit trees. *Applied Engineering in Agriculture* 29(2): 155-160.

Xiao, C., L. Zheng and H. Sun. 2014. Estimation of the Apple Flowers Based on Aerial Multispectral Image. In *2014 Montreal, Quebec Canada July 13–July 16, 2014,* 1. American Society of Agricultural and Biological Engineers.