Computers and Electronics in Agriculture 155 (2018) 386–393

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

Branch detection for apple trees trained in fruiting wall architecture using depth features and Regions-Convolutional Neural Network (R-CNN)

Jing Zhang[a,c], Long He[d], Manoj Karkee[a,b], Qin Zhang[a], Xin Zhang[a], Zongmei Gao[a]

[a] The Center for Precision and Automated Agricultural Systems, Washington State University, USA
[b] Department of Biological Systems Engineering, Washington State University, USA
[c] College of Engineering, China Agricultural University, China
[d] Department of Agricultural and Biological Engineering, Pennsylvania State University, USA

ARTICLE INFO

Keywords:
Branch detection
Branch skeleton fitting
Shake-and-catch apple harvesting
R-CNN
Depth features

ABSTRACT

Due to the rising cost and decreasing availability of labor, manual picking is becoming an increasing challenge for apple growers. A targeted shake-and-catch apple harvesting technique is being developed at Washington State University to address this challenge. The performance and productivity of such a harvesting technique can be increased greatly if the shaking process is automated. The first step toward automated shaking is the detection and localization of branches in apple tree canopies. A branch detection method was developed in this work for apple trees trained in a formal, fruiting wall architecture using depth features and a Regions-Convolutional Neural Network (R-CNN). Microsoft Kinect v2 was used to acquire RGB images and pseudo-color images, as well as depth images in natural orchard environment. The R-CNN was composed of an improved AlexNet network and was trained to detect apple tree branches using integrated pseudo-color and depth images for improved detection accuracy. The average recall and accuracy from the Pseudo-Color Image and Depth (PCI-D) method were 92% and 86% respectively when the R-CNN confidence level of the pseudo-color image was 50%. For comparison, when using the Pseudo-Color Image (PCI) method (without depth images), these averages were only 86% and 81%, respectively. Furthermore, the average correlation coefficient (r) between the fitting curves for branch skeletons using the PCI-D method and the fitting curves for ground-truth images was 0.91—another indicator that the PCI-D method performs better than the PCI method. In addition, the average accuracy of branch detection increased with both the PCI method and PCI-D method, since the sensor was closer to the canopy. This study demonstrates the great potential for using depth features in branch detection and skeleton estimation to develop effective shake-and-catch apple harvesting machines for use in formally trained apple orchards.

1. Introduction

Currently, fresh market apples are harvested by manual picking around the world. In Washington State alone, estimated seasonal laborers for apple harvesting were 77,100 in September 2015 and 58,800 in October 2015 (Forland and Sinkler, 2016). Growers spent $430,000 per km^2 for harvest labor in 2009; this increased to $560,000 per km^2 by 2014 (Delicious et al., 2009; Galinato et al. (2016)). Even worse, it is increasingly challenging for growers to find sufficient numbers of seasonal laborers during harvest. In order to reduce costs, improve harvest efficiency and decrease the risk of labor shortages, shake-and-catch harvesting machines are being developed and evaluated for fruit harvesting. Especially for the fruiting wall architecture, a targeted shake-and-catch harvesting technique could achieve substantially higher harvesting efficiency compared to robotic picking of individual fruit, and has achieved much improved fruit quality, showing potential for such techniques to be practical for some apple varieties (He et al., 2017; Sola-Guirado et al., 2017; Zhang and Karkee, 2016; Zhang et al., 2016).

One drawback is that the machines currently used for shake-and-catch harvesting are not fully automated; operators are required to engage the shaker on target branches. The first crucial step in developing a fully automated harvesting machine using the shake-and-catch technique is automated detection of tree branches in natural orchard environments using a machine vision system. Due to only a few parts of the branches will be visible (because occlusion by fruit and foliage) for some varieties of apple trees, the branch skeletons data in the dormant season can be used to provide desirable branch locations even when entire branches are occluded in harvest season. Some methods on fruit

E-mail addresses: cathy64882584@163.com (J. Zhang), luh378@psu.edu (L. He), manoj.karkee@wsu.edu (M. Karkee), qinzhang@wsu.edu (Q. Zhang), xin.zhang4@wsu.edu (X. Zhang), zongmei.gao@wsu.edu (Z. Gao).

https://doi.org/10.1016/j.compag.2018.10.029
Received 23 March 2018; Received in revised form 30 August 2018; Accepted 23 October 2018
Available online 02 November 2018

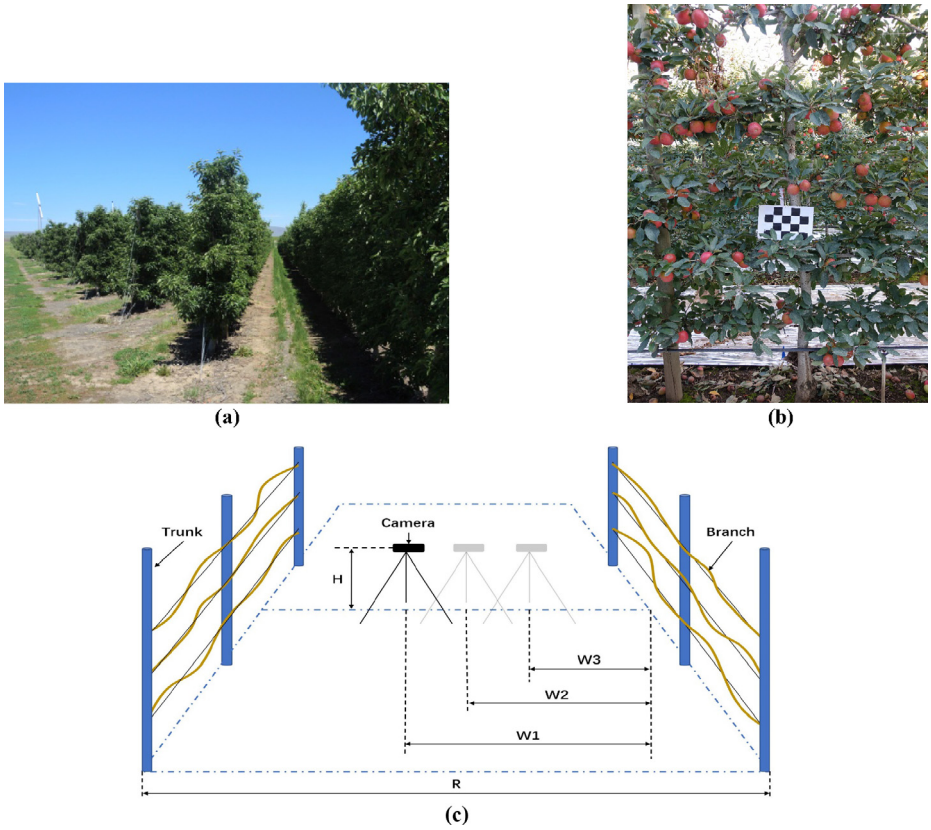


Fig. 1. (a) Experimental orchard used in this work; (b) Apple trees of the 'Jazz' variety during harvest season; and (c) Illustration of experimental image acquisition scheme.

tree branch detection have been developed in the past; for example, the three-dimensional (3D) medial axis thinning-based skeletonizing algorithm was used to reconstruct apple trees for mechanical pruning (Adhikari and Karkee, 2011). Other researchers combined 'stripe programming' with the contour model algorithm for 3D reconstruction of Chinese hickory trees (He et al., 2012). Both studies detected branches in relatively simple tree architectures and the test objects lacked flexibility, which caused the speed and environmental conditions of branch detection to be limited for practical adoption in real orchards. Another study used multi-class Support Vector Machines (SVM) and morphological operations to identify fruits and branches in citrus canopies (Qiang et al., 2014). The Bayesian classifier and curve fitting method were used for cherry branch detection with a resulting accuracy of 89.2% in a set of 141 test images with full-foliage canopy at night (Amatya et al., 2016). However, the clustering algorithm used for cherry branch detection also has some limitations, especially when the intensity and angle of external irradiation are variable, and when the obvious differences in tree architecture and branch size for cherry trees and apple trees are considered.

In recent years, deep learning approaches have been widely used in image classification, object detection and localization among other aspects (Hinton and Salakhutdinov, 2006; Kamilaris and Prenafeta-Boldú, 2018; Russakovsky et al., 2015). In various deep learning detection approaches, Regions-Convolutional Neural Network (R-CNN) is one of the most widely used methods for object detection. (Girshick et al., 2014). Although deep learning has been extensively used in many fields such as face recognition and semantic segmentation (Ranjan et al., 2016), the application in agricultural automation and robotics, especially for fruit and branch detection, is still limited. One such effort was reported by Bargoti and Underwood (2017) in which the researchers addressed the specific constraints imposed by detection of mangoes, almonds and apples in large scale orchard data using a deep learning detector. However, due to the differences between the detected object and the application environment, it is difficult for this method to be used without modification for the detection of apple tree branches in the orchard. Adopting deep learning techniques and knowledge could be one of the key approaches for solving challenging problems in agriculture.

The primary contribution of this study was deploying depth features and R-CNN technique to achieve the branch detection and improve the detection accuracy; the establishment of branch skeleton equations could be used to the reference when the branches are completely occluded by fruit and foliage; the results are expected to provide baseline technology for developing fully automated shake-and-catch harvesting machinery. The specific objectives of this study are to: (i) detect and locate the branches of apple trees from the pseudo-color and depth images using R-CNN in natural orchard environment; and (ii) establish the corresponding mathematical models that represent the skeleton of the detected branches.

2. Materials and methods

This study was focused on detecting branches of apple trees trained in a fruiting wall architecture. The images of apple trees were acquired in the real orchard from varying distances using a Microsoft Kinect v2 camera. A set of sensor data including RGB images, pseudo-color images and depth images were acquired. Among them, the pseudo-color images and depth images were selected as the inputs to the R-CNN for training and learning. After using a confidence filter to scan the detection results of R-CNN, the results were further improved by objective boundary optimization and the centroid estimation algorithm. The skeleton fitting functions were used to represent the branch locations in the RGB images. This study verified the performance and feasibility of the proposed method. Detailed explanations of the method will be

provided in the following sections.

2.1. Image acquisition

The complex background factors and changes of illumination in the natural orchard environment always make the object detection difficult for traditional image processing techniques to achieve desired accuracy. However, using the depth features, it is possible to not only estimate the specific space location of objects, but also to remove most background interference by limiting the detection range of depth. Therefore, this study used a sensor (Kinect v2, Microsoft Inc, Seattle, WA) to build an image acquisition system and capture color (RGB) and depth images. The sensor used the principle of Time of Flight (ToF) to acquire depth information with an operating range of 0.5–4.5 m, and average error in depth measurement ranging from 2 to 4 mm within the operating range of 3.5 m (Yang et al., 2015). The Kinect v2 sensor is highly stabile and cost-effective and also offers a relatively shorter development time because there is a good research foundation (Nissimov et al., 2015; Song and Xiao, 2016).

In this study, all the images were acquired in a commercial apple orchard near Prosser, WA (Fig. 1a). As almost entire branches were occluded by foliage or fruits for the apple trees ('Jazz' variety) during the harvest season (Fig. 1b), we picked the dormant season to acquire the images for the following reasons: (i) the size and position of the main branches and trunks will not have obvious changes from dormant season to harvest season; (ii) the branch skeleton location information could be captured and stored by computer in the dormant season, so that the auto-guided harvesting machine can pick target branches from the database even when the branches are completely occluded. Fig. 1c illustrates the schematic for image acquisition: (i) the row spacing R was ~260 cm; (ii) the Kinect v2 was mounted on a tripod to maintain a vertical distance, H, of ~138 cm from the ground; and (iii) the 2nd to 4th layers of apple tree branches were used as the targets for detection. The images were acquired from three different distances; viz. W1, W2 and W3, which were 182 cm, 159 cm and 108 cm, respectively. The objects of interest in these images were only the primary branches that are candidates for shaking during shake-and-catch harvesting processing between two adjacent apple tree trunks. In total, 270 sets of images were taken from 90 target sections of selected apple trees.

As mentioned before, two distinct types of images were obtained from each location of the Kinect v2. Fig. 2 shows two examples of those images: (i) an original RGB image (at a resolution of 1344 × 756; Fig. 2a), which is difficult to use with the traditional image processing method to segment the apple tree branches out of the image; and (ii) a pseudo-color image (at a resolution of 512 × 424; Fig. 2b). In addition, a depth image with only depth channel was obtained at a resolution of 512 × 424. The pseudo-color image and depth image are two different images with certain correlations. The depth images are composed of the actual value of depth at each pixel (Hanselman and Littlefield, 2001). The pseudo-color images are derived from the depth images by mapping each depth value to a color value, which make the depth image visible. In this study, a jet color model was used to convert the depth images to the pseudo-color images. The desired depth range to canopy objects was maintained between 0.5 and 4 m and the images were taken on cloudy days.

2.2. R-CNN and image training

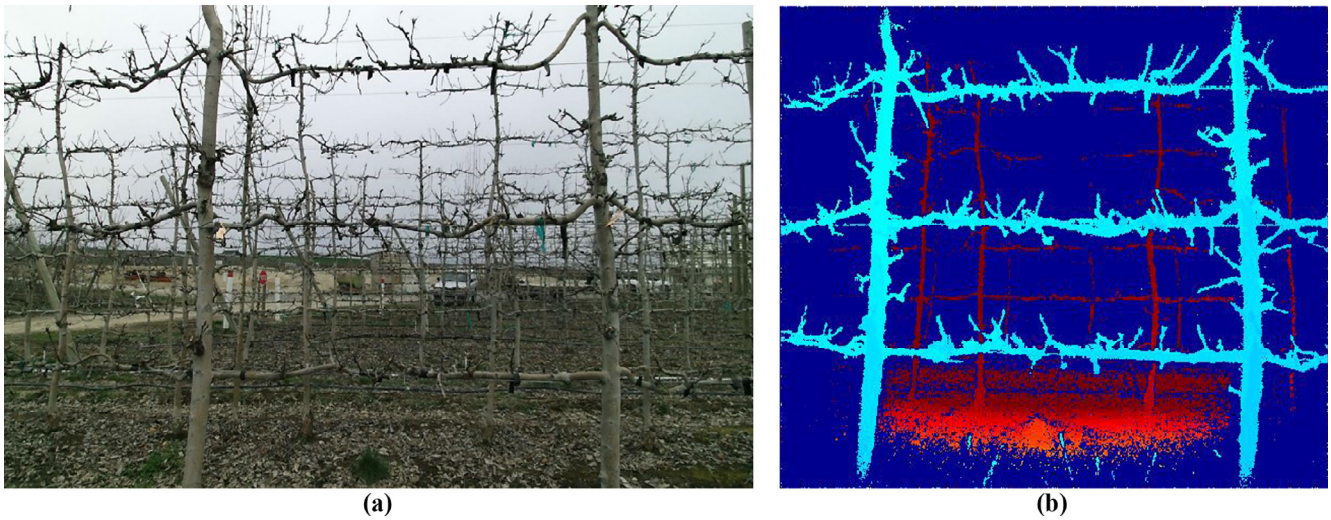
2.2.1. Implementation of R-CNN

The object detection method (R-CNN) includes both identification of categories and the localization (or positioning) of objects in images, which can be represented by bounding boxes containing the objects. According to the selective search method (Dietz et al., 2006), 2000 region proposals were searched by R-CNN from the input image, but these region proposals are hard to match perfectly with the objects which are labeled manually (Uijlings et al., 2013). Therefore, an Intersection over Union (IoU) parameter was used to evaluate the accuracy of these regional proposals (Long et al., 2015). IoU is the overlap ratio between the Predicted Box (PB) and the Ground-truth Box (GB), as shown in Eq. (1).

$$IoU = \frac{PB \cap GB}{PB \cup GB} \tag{1}$$

If the IoU value is greater than a set threshold, the region proposal is considered to contain the object, otherwise it is classified as background. Because only 270 ground-truth images were manually labeled in this study, it is necessary to use the pre-training network for transfer learning, an approach that also reduces development and training time (Sermanet et al., 2013). In the pre-training section of this study, the network was pre-trained using a subset of the ImageNet dataset (Deng et al., 2009) which contains more than a million images and classifies those images into 1000 object categories. During the fine-tuning, the fully connected layer and classification layer of the pre-trained network need to be replaced with layers based on the ground-truth data of apple tree branches; after this substitution, the pre-trained network can be used to train the new dataset after the configuration of training parameters.

The pre-trained AlexNet developed by Krizhevsky et al. (2012) was selected as the network structure of the R-CNN in this study. AlexNet



(a) (b)

Fig. 2. (a) An example RGB image acquired with Kinect v2; and (b) An example pseudo-color image acquired with Kinect v2.

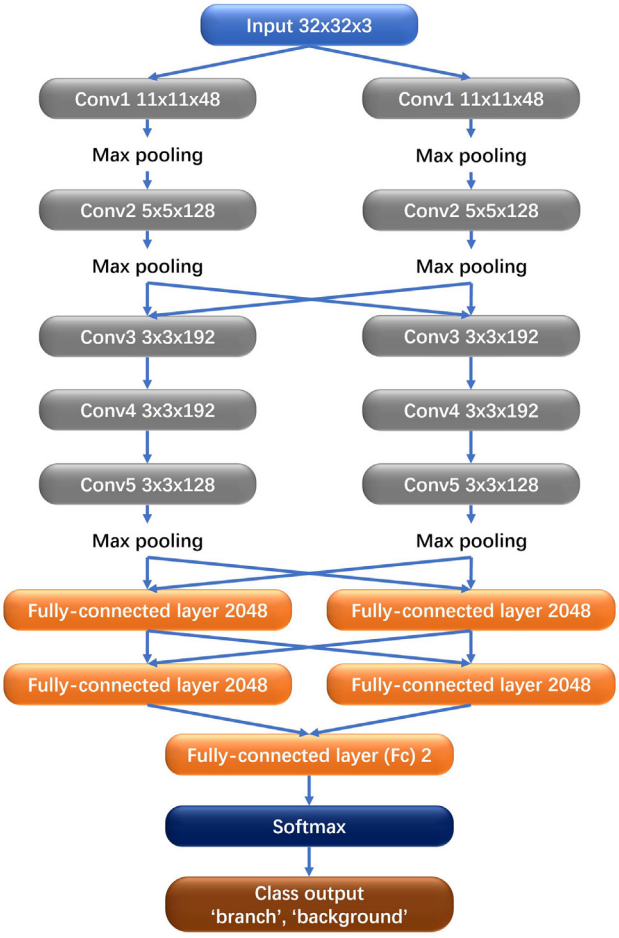


Fig. 3. Trained network with ground-truth images of apple tree branches based on the AlexNet; The trained network also followed the same splitting and concatenate principle with Alexnet; The size of max pooling layers was 2 × 2 with a stride of 1 pixel; Conv – Convolution.

was the champion of the 2012 ImageNet Large Scale Visual Recognition Challenge (ILSVRC2012) and is also the foundation of many other excellent network structures. Fig. 3 illustrates the trained network after fine-tuning with our own dataset. In Fig. 3: (i) the input layer pre-processed the RGB images to create candidate regions of size 32 × 32 × 3; (ii) following to the original implementation of AlexNet, the Convolution (Conv) layers split the input evenly into two sections along the kernel number, and then concatenated the two resulting sections together to produce the output. For example, in the Conv1 layer, the input to the layer had 96 kernels, and was split into two sections with 48 kernels; (iii) the Rectified Linear Unit (ReLU) activation was applied to the output of every convolutional and fully-connected layer. The Local Response Normalization (LRN) function was also used after the ReLU between Conv1 and Conv2, as well as Conv2 and Conv3 (Nair and Hinton, 2010); (iv) the size of max pooling layers was 2 × 2 with a stride of 1 pixel because the size of input region was different from the pre-trained AlexNet (Masci et al., 2011); (v) a new fully-connected layer (Fc) was added to the network and set to have the same size as the number of classes in our own dataset according to the transfer learning method; and (vi) the confidence that the bounding boxes were considered as the apple tree branch was given by the Softmax layer. A large confidence also means a less feature loss (Liu et al., 2016).

2.2.2. *Image training and hyperparameter configuration*

The training performance of R-CNN plays a dominant role for the detection accuracy of objects. This study used the pseudo-color image and the depth image as two separate sections for training. In the pseudo-color image section, 201 ground-truth images of apple tree branches were used as training sample for the R-CNN and the remaining 69 ground-truth images were used for testing. Fig. 4 shows a series of examples of ground-truth images with the same branches but different sensor distances (182 cm – Fig. 4a; 159 cm – Fig. 4b; and 108 cm – Fig. 4c). As shown in Fig. 4, each branch was manually labeled using multiple bounding boxes; in this manner we also provided a foundation for the next step that established a corresponding equation for the branch skeleton. In the depth image section, each depth image was paired with a pseudo-color image; therefore, the manual labeling boxes of the pseudo-color images were directly mapped onto the depth images, and the depth images were trained in the same network as the pseudo-color images.

During R-CNN training, different hyperparameters lead to different training results. In this study, although the pseudo-color images and the depth images were used as two separate sections, they were trained with the same training hyperparameters. After many trial-and-error training runs, the global learning rate of R-CNN was set to 0.0001, and the size of the mini-batch used in each training iteration was set to 256. The training was continued for 100 epochs and the global learning rate remained unchanged throughout the training process. Moreover, because the object features in each bounding box were quite similar and there were many bounding boxes in each of the ground-truth images, the region proposal was considered positive when its IoU was ≥0.3; otherwise it was negative. After each training, the training result and

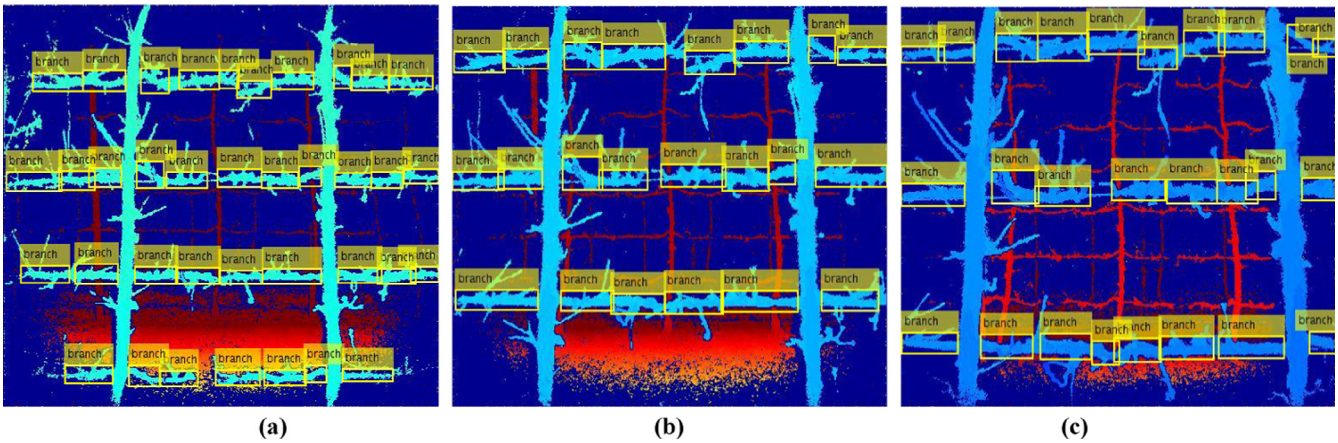


Fig. 4. Example ground-truth images with the same object in each image collected from three different distances; (a) W1 (182 cm); (b) W2 (159 cm); and c) W3 (108 cm).

corresponding hyperparameter configuration were saved to compare with the result from the next training.

2.2.3. Assessment parameters of training results

In this study, the training results were evaluated by using the recall and accuracy parameters of each image. The recall (also known as sensitivity) is defined as the fraction of ground-truth results that have been detected over the total amount of ground-truth results. Therefore, a greater recall value means that fewer objects are missed within the same precision. The recall is particularly important for the branch detection method in this study, especially for creating a model to fit the entire branch skeleton. Complete detection of branches is essential to achieve higher model fitting accuracy.

Accuracy is also an important parameter for evaluating the results of object detection, which can contain both random errors and systematic errors. High accuracy requires both high precision and high trueness (ISO-5725-1, 1994). Therefore, the accuracy can be a more comprehensive measure for evaluating the results of branch detection in this study. In addition, a confidence level will be provided by R-CNN for each detection result (He et al., 2015). The confidence level is typically used to reflect the probability of True Positive (TP) for the detection result in most cases.

2.3. Branch detection

There were two methods used to detect the apple tree branches, namely the Pseudo-Color Image (PCI) detection method and the Pseudo-Color Image and Depth (PCI-D) detection method. In the PCI method, the pseudo-color image was input directly into the R-CNN for branch detection. In the PCI-D method, however, the R-CNN was used to detect branches using both pseudo-color images and depth images separately; then the results from the two types of images were fused into one image using the confidence filter. The main reason for enhanced branch detection using the combination of pseudo-color images and depth images is that these two types of images show distinctive features of branches at the pixel level.

2.3.1. The overall process of branch detection

The flowchart of the overall technique is shown in Fig. 5. Either the PCI or the PCI-D method was selected for branch detection. If the PCI method was selected, only pseudo-color images were used as inputs for the R-CNN. On the other hand, if PCI-D method was selected, then both pseudo-color images and depth images were used as inputs for the R-CNN. The R-CNN detection results from the pseudo-color images and the depth images were both used as the outputs. Then a boundary optimization algorithm was used, and the centroid of each object region was extracted. Through the coordinate mapping, the detected bounding boxes and centroids from the pseudo-color image were mapped onto the corresponding original RGB images. Finally, all the centroid coordinates were used to fit the skeleton equation for branches in RGB images.

2.3.2. Boundary optimization and coordinate mapping

A boundary optimization algorithm was used to improve the size of the bounding boxes (branch regions in this study) and the accuracy of centroid estimation. In some cases, the size of the object region obtained by R-CNN was larger than the actual object in the image, a result which might lead to errors in estimating the centroid position. Therefore, use of a boundary optimization algorithm is practical and necessary for this study. In the pseudo-color images, the color difference between the branches and the background is obvious because a greater depth difference is represented by a more significant color difference. Using this color feature, the bounding boxes of object regions detected by R-CNN were further optimized.

The boundary optimization algorithm consisted of the two steps:

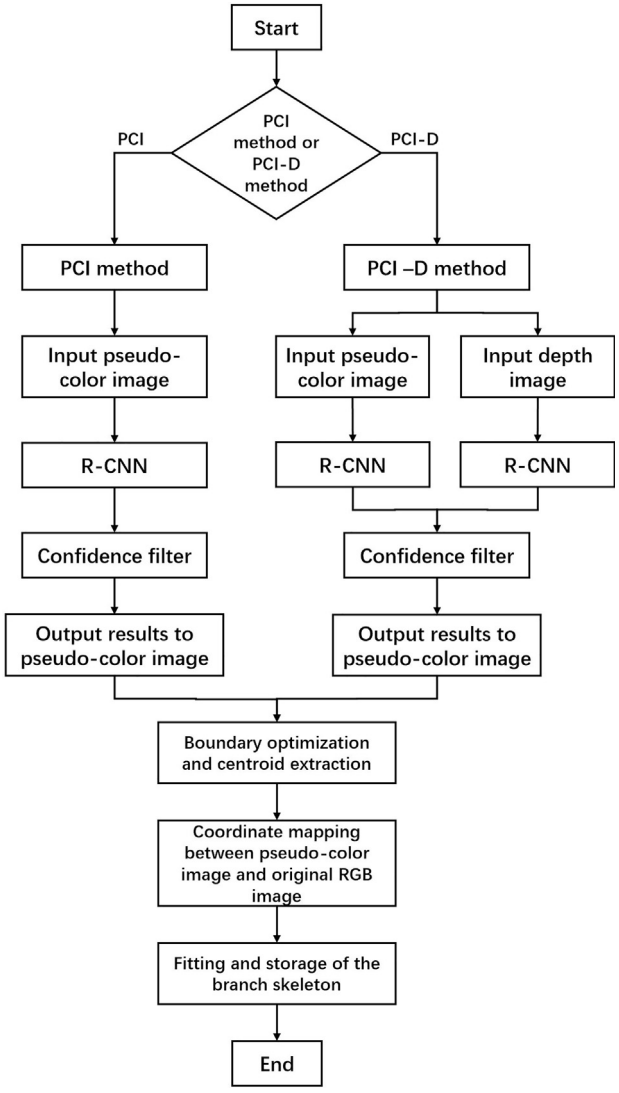


Fig. 5. The flow chart of branch detection technique.

(1) Step 1: A matrix P (Eq. (2)) is used to represent the coordinates of each pixel in the object bounding box, and the matrix has the same number of rows and columns as the corresponding bounding box. The pixel values at those coordinates are represented by matrix W (Eq. (3)).

$$P = \begin{bmatrix} (x_1, y_1) & \cdots & (x_i, y_1) \\ \vdots & \ddots & \vdots \\ (x_1, y_n) & \cdots & (x_i, y_n) \end{bmatrix} \tag{2}$$

Matrix W is the result of matrix P transformed by the function *impixel*. In Eq. (3), V_{11} to V_{in} are the RGB values that correspond to the coordinates (x_1, y_1) to (x_i, y_n), respectively.

$$W = \begin{bmatrix} V_{11} \\ V_{21} \\ \vdots \\ V_{in} \end{bmatrix} Q = \begin{bmatrix} V_{11} & \cdots & V_{1n} \\ \vdots & \ddots & \vdots \\ V_{i1} & \cdots & V_{i\times n} \end{bmatrix} Q^T = \begin{bmatrix} V_{11} & \cdots & V_{i1} \\ \vdots & \ddots & \vdots \\ V_{1n} & \cdots & V_{i\times n} \end{bmatrix} \tag{3}$$

(2) Step 2: All the pixels in the object regions were visited to find the RGB information using the matrix Q^T (Eq. (3)). For each pixel in the object regions, the sum of R, G and B components was computed; if the sum total was above 600, the pixel was considered part of the object, and if it was below this threshold, the pixel was considered part of the background. For all pixels found from Q^T that were

considered objects, the minimum pixel coordinate was used as the upper left coordinate of the optimized object region, and the maximum coordinate was used as the lower right coordinate of the rectangular. This process leads to an optimized boundary of the object region replacing the original boundary detected by R-CNN.

In this study, the pseudo-color and RGB images had different pixel resolutions (512 × 424 for pseudo-color image versus 1344 × 756 for RGB images). Therefore, it was necessary to map the detection results of apple tree branches from pseudo-color images onto the RGB images using a coordinate mapping algorithm (Terven and Córdova-Esparza, 2016). The detection results for branches became more intuitive and conducive for human usage after use of the coordinate mapping algorithm. The algorithm also allowed the skeleton equations of branches to be fitted in the RGB images, making the contrast between the ground-truth branch skeletons and the fitted branch skeletons clearly visible.

In this study, an Intel i7-6700HQ CPU and NVIDIA GTX960M GPU constituted the hardware configuration of the computer. The branch detection program was developed in MATLAB 2017a environment.

3. Results and analysis

3.1. Branch detection with different sensor positions

The average accuracy of branch detection with different sensor positions was compared in this study. The confidence filters for pseudo-color images and depth images were set to 50% and 70%, respectively. Based on sensor positions (W1, W2, and W3), the test images were divided into 'Far', 'Medium' and 'Close' distance groups respectively. The average accuracy of branch detection with PCI and PCI-D methods is presented in Table 1.

As shown in Table 1, with a decrease in distance between the sensor and the apple tree canopy, the average branch detection accuracy increased for both the PCI and PCI-D methods. For all evaluated sensor positions, the average accuracy of the PCI-D method was always higher than the average accuracy of the PCI method. Notably, when the sensor position was closest to the apple trees, the average accuracy of PCI-D method was the highest (90.8%). This result indicates that the sensor should be set close to the apple tree canopies (108 cm was the closest distance tested) while ensuring that the targeted tree branches are visible in the pseudo-color image.

3.2. Recall and accuracy of branch detection with different confidence levels

Besides the investigation of branch detection accuracy with varying sensor positions, the recall and accuracy of branch detection results were also evaluated based on different confidence levels. A higher confidence means the bounding box estimated by R-CNN is more likely to be TP. This experiment attempted to find optimal confidence levels for R-CNN results so that higher levels of recall and accuracy could be achieved. During the experiment, the confidence level of R-CNN was set at a fixed rate of 70.0% for the depth images because lower confidence levels may cause excessive overlaps of the bounding box. The test images used in this experiment included 69 pseudo-color images and 69 depth images. As shown in Fig. 6, five different confidence levels of R-CNN were used. Among these five levels, the average recall and the average accuracy with PCI and PCI-D methods were tested individually to find the optimal confidence level for branch detection.

Table 1
Average accuracy of branch detection with varying sensor positions.

Sensor positions	Average accuracy of branch detection	
	PCI method	PCI-D method
Far (W1)	76.1%	78.5%
Medium (W2)	81.3%	87.2%
Close (W3)	85.3%	90.8%

As shown in Fig. 6a, as the confidence level of R-CNN increased, on average, recall decreased. When the R-CNN confidence level was 50.0%, both PCI and PCI-D methods achieved the highest average recall of 85.6% and 91.5%, respectively. For all confidence levels, the PCI-D method achieved higher average recall compared to the PCI method. Particularly, the average recall with the PCI-D method was 5.9% and 5.1% higher than that of the PCI method when the confidence levels were 50.0% and 60.0% respectively. As shown in Fig. 6b, when the R-CNN confidence level decreased, branch detection accuracy increased on average. With decreasing R-CNN confidence, numbers of both False Negative (FN) and TP detections decreased. However, FN detections decreased more rapidly than TP did, in the experiment, leading to increased detection accuracy. Over all R-CNN confidence levels, the average accuracy with the PCI-D method was higher than that with the PCI method. When the confidence level of R-CNN was set at 50.0%, the PCI-D method achieved the highest average accuracy of 85.5%. The results of recall and accuracy indicate that the average accuracy of branch detection has been improved; meanwhile, the sensitivity of the algorithm was also enhanced when the confidence level of R-CNN was set at 50.0%. In addition, the fusion of pseudo-color images and depth images to detect apple tree branches was more effective than using pseudo-color images alone when the R-CNN method was applied to object detection.

3.3. Branch skeleton fitting

It is important to obtain a continuous branch skeleton equation so that a shaker (part of the harvesting machine) can be engaged at any location on the branch to facilitate fruit removal. A polynomial fitting method was used to achieve the skeleton equation of the branches in this experiment. An example skeleton represented by polynomial equation is shown in Fig. 7. The figure demonstrates the entire process of skeleton fitting and compares the results of the PCI and PCI-D methods.

Due to the limitation of the sensor field-of-view, each image mainly focused on the 2nd to 4th layers of apple tree branches between two adjacent trunks. The size of some bounding boxes in the initial results (Fig. 7a and c respectively) have been modified through the boundary optimization algorithm represented by Fig. 7b and d. The red dots (Fig. 7b and d respectively) show the centroid position of each bounding box; these centroid positions were then used to fit the branch skeleton equations. Using a coordinate mapping technique, all the bounding boxes generated by R-CNN, along with the centroid coordinates, were mapped from each pseudo-color image to the corresponding RGB image as shown in Fig. 7e and f. Note that no bounding boxes are shown in Fig. 7f in order to display the fitting results of the branch skeleton more clearly. The red curves (Fig. 7e and f, respectively) show the branch skeletons as represented by corresponding polynomial equations (5-order polynomials in this example). The order of the polynomial can be varied to obtain reasonable accuracy.

The blue curves (Fig. 7e and 7f respectively) represent the branch skeleton obtained using the ground-truth images (similar to Fig. 4). These fitting curves were used as references to assess the accuracy of branch skeleton fitting. The correlation coefficient (r) was used to describe the similarity between the red curve (detection skeleton) and the blue curve (reference skeleton). In 69 test images, the average correlation coefficient (r) for branch skeletons estimated with the PCI method and the PCI-D method were 0.86 and 0.91, respectively. The experimental results showed that the PCI-D method improved the branch fitting accuracy compared with the PCI method for all the test images. In addition, there were small twigs on the apple tree branches that negatively influenced the determination of centroid location in the pseudo-color images, thereby causing a few centroid locations to be

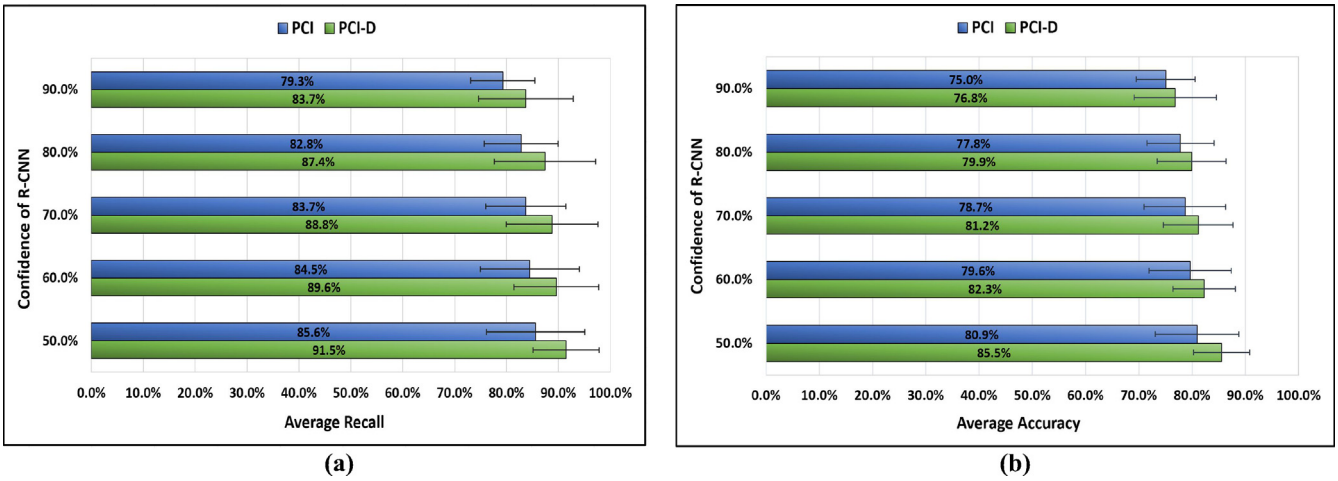


Fig. 6. (a) Bar graph showing average recall with different confidence level of R-CNN resulted in by PCI and PCI-D methods; and (b) Bar graph showing average accuracy with the same experiment.

represented erroneously and thus influencing the accuracy of the branch skeleton estimation.

This study showed promising results in detecting, representing and localizing branches of apple trees during the dormant season in the natural orchard environment. This is a crucial and necessary first step toward developing branch detection techniques and can be a reference when the entire branches are occluded during the harvest season. The accuracy of branch detection and skeleton fitting, as well as the computational speed of the approach presented in this work, may need further improvement to make this method practical enough for automated shake-and-catch apple harvesting.

4. Conclusions

The purposes of this study were to devise a reliable and robust machine vision method for detecting apple tree branches in real orchard and to develop mathematical models for representing the branch skeleton. The depth features of branches were used as input to the R-CNN with the goal of improving the recall and accuracy of branch detection and the accuracy of branch skeleton fitting. The following specific conclusions can be drawn from this study:

1. The accuracy of branch detection increased as the sensing system was moved closer to tree canopies. It was also found that for all three tested sensor locations (182 cm, 159 cm and 108 cm, respectively), the average accuracy of branch detection with the PCI-D method was always higher than with the PCI method.
2. The average recall and accuracy of branch detection were improved with decreasing R-CNN confidence levels. When the R-CNN confidence level was 50.0% for pseudo-color images, the PCI-D method achieved the highest average recall and accuracy (91.5% and 85.5%, respectively), which were 5.9% and 4.6% higher than with

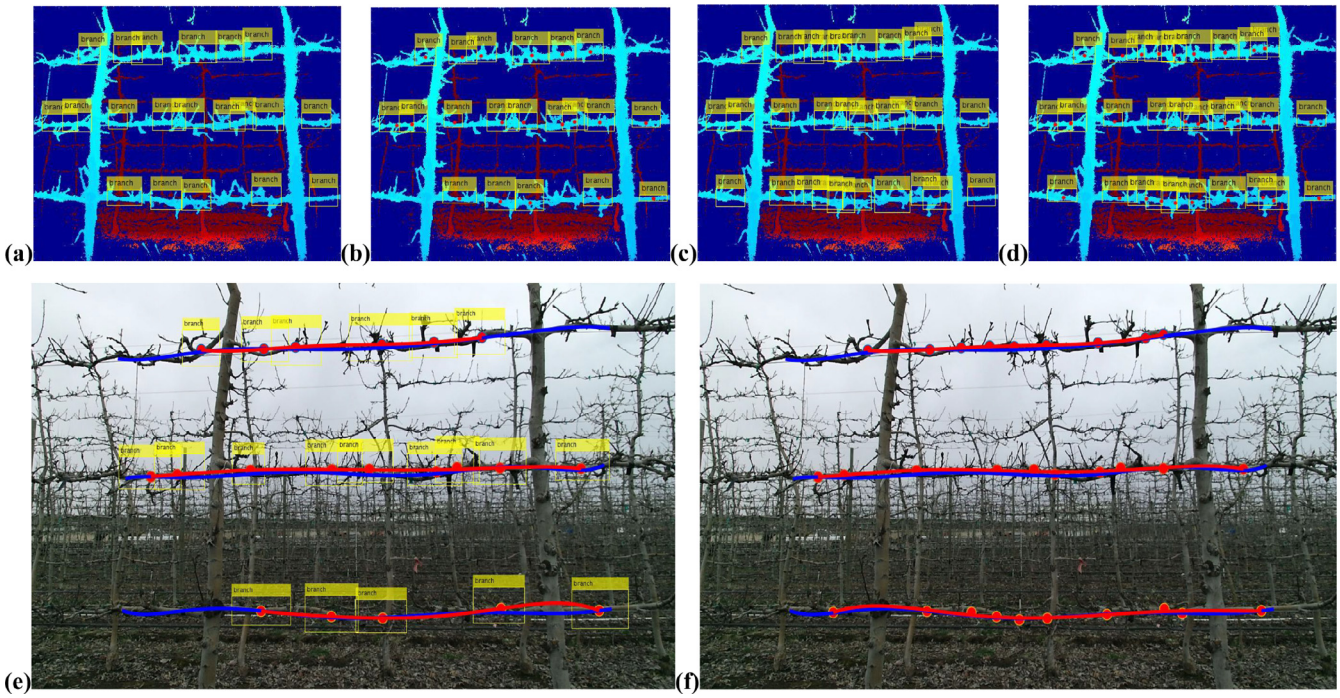


Fig. 7. (a) Branch detection results using the PCI method; (b) Results of bounding box optimization and centroid estimation with the PCI method; (c) Branch detection results using the PCI-D method; (d) Results of bounding box optimization and centroid estimation with the PCI-D method; (e) Branch skeleton fitting with the branch detection results from the PCI method; and (f) Branch skeleton fitting with the branch detection results from the PCI-D method (without bounding box).

the PCI method. The results indicate that integration of pseudo-color and depth images is more effective in detecting apple tree branches than using only pseudo-color images.

3. The branch skeletons were fitted by the polynomial equations. As expected from the branch detection results, the correlation coefficient (r) values (accuracy of branch skeleton estimation) detected by the PCI-D method and the PCI method were 0.91 and 0.86, respectively.

The results, in general, indicate that the fusion of pseudo-color images and depth images lead to better performance in detecting branches, locating them and fitting their skeletons. Further research can combine a more efficient and accurate Faster R-CNN method to detect apple tree branches; additionally, more classifications of objects can be recognized including apples and trunks. The navigation system and the attitude sensor can also be integrated in the future to record the actual sensor location and sensor movement path. This study has provided a foundation and highlighted the potential for automatically detecting the shaking locations on the apple tree branches, which would be crucial for the success of an automatic shake-and-catch harvesting machine.

Acknowledgments

This research was supported in part by USDA Hatch and Multistate Project Funds (Accession Nos. 1005756 and 1001246), a USDA National Institute for Food and Agriculture competitive grant (Accession No. 1005200), and the Washington State University (WSU) Agricultural Research Center. The China Scholarship Council (CSC) sponsored Jing Zhang in conducting collaborative PhD dissertation research at the WSU Center for Precision and Automated Agricultural Systems (CPAAS). Any opinions, findings, conclusions, or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the USDA and Washington State University.

References

Adhikari, B., Karkee, M., 2011. 3d Reconstruction of Apple Trees for Mechanical Pruning, ASABE Annu. Int. Meet 7004.

Amatya, S., Karkee, M., Gongal, A., Zhang, Q., Whiting, M.D., 2016. Detection of cherry tree branches with full foliage in planar architecture for automated sweet-cherry harvesting. Biosyst. Eng. 146, 3–15.

Bargoti, S., Underwood, J.P., 2017. Image segmentation for fruit detection and yield estimation in apple orchards. J. Field Robot.

Delicious, R., Smith, G., Delicious, G., Pink, C., 2009. 2009 Cost estimates of establishing and producing gala apples in Washington. Red 2008, 2009.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database. In: Computer Vision and Pattern Recognition, 2009, CVPR 2009, IEEE Conference on IEEE.

Dietz, H., Berkemeier, F., Bertz, M., Rief, M., 2006. Anisotropic deformation response of single protein molecules. Proc. Natl. Acad. Sci. 103 (34), 12724–12728.

Forland, C., Sinkler, A., 2016. Labor Market and Performance Analysis.

Galinato, S. P., Gallardo, R.K., Hong, Y.A., 2016. 2014 cost estimates of establishing, producing, and packing organic Gala apples in Washington.

Girshick, R., Donahue, J., Darrell, T., Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern recognition.

Hanselman, D., Littlefield, B., 2001. Mastering MATLAB 6: A Comprehensive Tutorial and Reference. Pearson.

He, L., Du, X., Qiu, G., Wu, C., 2012. 3D reconstruction of Chinese hickory trees for mechanical harvest. In 2012 Dallas, Texas, July 29-August 1, 2012. American Society of Agricultural and Biological Engineers.

He, L., Fu, H., Sun, D., Karkee, M., Zhang, Q., 2017. Shake-and-catch harvesting for fresh market apples in trellis-trained trees. Trans. ASABE 60 (2), 353–360.

He, M., Zhang, S., Mao, H., Jin, L., 2015. Recognition confidence analysis of handwritten Chinese character with CNN. In: Document Analysis and Recognition (ICDAR), 2015 13th International Conference on. IEEE.

Hinton, G.E., Salakhutdinov, R.R., 2006. Reducing the dimensionality of data with neural networks. Science 313 (5786), 504–507.

ISO-5725-1, 1994. Accuracy (trueness and precision) of measurement methods and results. Available at: https://www.iso.org/obp/ui/#iso:std:iso:5725:-1:ed-1:en.

Kamilaris, A., Prenafeta-Boldú, F.X., 2018. Deep learning in agriculture: a survey. Comput. Electron. Agric. 147, 70–90.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems.

Liu, W., Wen, Y., Yu, Z., Yang, M., 2016. Large-Margin Softmax Loss for Convolutional Neural Networks. In ICML.

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Masci, J., Meier, U., Cireşan, D., Schmidhuber, J., 2011. Stacked convolutional auto-encoders for hierarchical feature extraction. Artif. Neural Networks Mach. Learning–ICANN 52–59.

Nair, V., Hinton, G.E., 2010. Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th International Conference on Machine Learning (ICML-10).

Nissimov, S., Goldberger, J., Alchanatis, V., 2015. Obstacle detection in a greenhouse environment using the Kinect sensor. Comput. Electron. Agric. 113, 104–115.

Qiang, L., Jianrong, C., Bin, L., Lie, D., Yajing, Z., 2014. Identification of fruit and branch in natural scenes for citrus harvesting robot using machine vision and support vector machine. Int. J. Agric. Biol. Eng. 7 (2), 115–121.

Ranjan, R., Patel, V.M., Chellappa, R., 2016. Hyperface: a deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. arXiv preprint arXiv:1603.01249.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., 2015. Imagenet large scale visual recognition challenge. Int. J. Comput. Vision 115 (3), 211–252.

Sermanet, P., Kavukcuoglu, K., Chintala, S., LeCun, Y., 2013. Pedestrian detection with unsupervised multi-stage feature learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Sola-Guirado, R.R., Ceular-Ortiz, D., Gil-Ribes, J.A., 2017. Automated system for real time tree canopy contact with canopy shakers. Comput. Electron. Agric. 143, 139–148.

Song, S., Xiao, J., 2016. Deep sliding shapes for amodal 3D object detection in RGB-D images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Terven, J.R., Córdova-Esparza, D.M., 2016. Kin2. A Kinect 2 toolbox for MATLAB. In: Science of Computer Programming.

Uijlings, J.R., Van De Sande, K.E., Gevers, T., Smeulders, A.W., 2013. Selective search for object recognition. Int. J. Comput. Vision 104 (2), 154–171.

Yang, L., Zhang, L., Dong, H., Alelaiwi, A., El Saddik, A., 2015. Evaluating and improving the depth accuracy of Kinect for Windows v2. IEEE Sens. J. 15 (8), 4275–4285.

Zhang, Q., Karkee, M., 2016. Fully Automated Tree Fruit Harvesting.

Zhang, Z., Heinemann, P.H., Liu, J., Baugher, T.A., Schupp, J. R., 2016. The Development of Mechanical Apple Harvesting Technology: A Review.