Intel Serv Robotics (2010) 3:245–262
DOI 10.1007/s11370-010-0074-3

SPECIAL ISSUE

Comprehensive Automation for Specialty Crops: Year 1 results and lessons learned

Sanjiv Singh · Marcel Bergerman · Jillian Cannons · Benjamin Grocholsky · Bradley Hamner · German Holguin · Larry Hull · Vincent Jones · George Kantor · Harvey Koselka · Guiqin Li · James Owen · Johnny Park · Wenfan Shi · James Teza

Received: 13 February 2010 / Accepted: 9 July 2010 / Published online: 11 August 2010

Abstract Comprehensive Automation for Specialty Crops is a project focused on the needs of the specialty crops sector, with a focus on apples and nursery trees. The project's main thrusts are the integration of robotics technology and plant science; understanding and overcoming socio-economic barriers to technology adoption; and making the results available to growers and stakeholders through a nationwide outreach program. In this article, we present the results obtained and lessons learned in the first year of the project with a reconfigurable mobility infrastructure for autonomous farm driving. We then present sensor systems developed to enable three real-world agricultural applications—insect monitoring, crop load scouting, and caliper measurement—and discuss how they can be deployed autonomously to yield increased production efficiency and reduced labor costs.

Keywords Specialty crops · Reconfigurable mobility · Crop intelligence · Insect monitoring · Crop load estimation · Caliper measurement

S. Singh · M. Bergerman (✉) · B. Grocholsky · B. Hamner · G. Kantor · W. Shi · J. Teza
Carnegie Mellon University, Pittsburgh, USA
e-mail: marcel@cmu.edu

J. Cannons · H. Koselka
Vision Robotics, San Diego, USA

G. Holguin · G. Li · J. Park
Purdue University, West Lafayette, USA

L. Hull
Pennsylvania State University, Biglerville, USA

V. Jones
Washington State University, Wenatchee, USA

J. Owen
Oregon State University, Aurora, USA

1 Introduction

Specialty crops are defined in the US as fruits, vegetables, tree nuts, dried fruits, nursery crops, and floriculture. Their market value in 2007 neared $50 billion, or almost 17% of the entire US agricultural market value, up from $41.2 billion in 2002—an annual growth of 3.9% [15]. In 2007, the five largest fruit and tree nut crops (grapes, apples, almonds, strawberries, and oranges) brought $11.0 billion in cash receipts to farmers [8]. Fruit and tree nut production alone generate about 13% of all farm cash receipts in the country.

Especially in the tree fruit industry, labor represents a large percentage of production costs (Fig. 1) and automation is not as widely available as in program crops, such as corn, soy, and wheat. Comprehensive Automation for Specialty Crops (CASC) aims at developing technologies and methods to improve production efficiency and reduce labor costs in the apple and nursery tree industries. The project is based on three main pillars: integration of robotics technology and plant science, overcoming socio-economic barriers that prevent or delay technology adoption by growers, and making the results available to growers and stakeholders through a nationwide outreach program. For a general overview of the project's goals, we refer the reader to [10].

Central to our work is the development and deployment of automated prime movers, or APMs—a family of reconfigurable robotic vehicles that can autonomously drive in fruit production environments (orchards, groves, vineyards, etc.) and nurseries. The APMs can carry sensors, instruments, farm implements, and even people to automate or augment production operations, including:

- harvesting, pruning, spraying, and mowing;
- plant inspection for stress, disease, and insect detection;

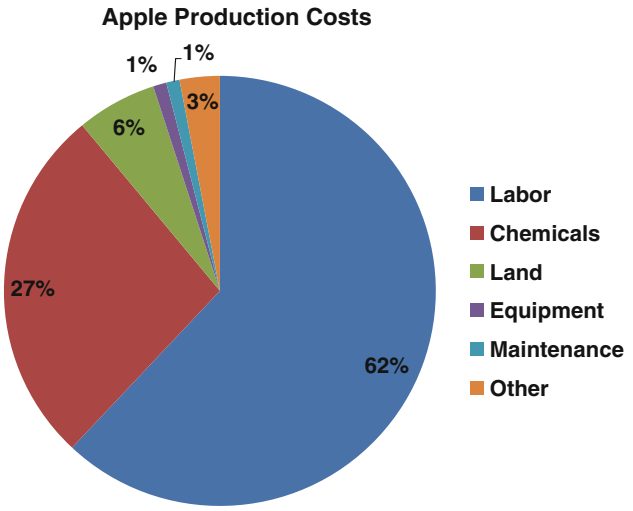

Fig. 1 Distribution of costs in apple production (adapted from [14]). In general, more than half of the cost of tree fruit production can be attributed to labor

- crop load estimation, caliper measurement, tree counting, etc.

We expect APMs to become to the specialty crops industry what the personal computers are to businesses worldwide: multifunctional, modular, expandable, accessible systems of various sizes and shapes, ranging from small electric utility vehicles to large over-the-row tractors and platforms, with a common autonomy infrastructure.

This article is composed of two main sections. In Sect. 1, we describe the development of the first vehicle in the APM family, based on the Toro eWorkman utility vehicle, and present the results obtained in more than 130 km of autonomous orchard traversal. In Sect. 2, we describe some of the sensors we are developing to automate plant and crop data collection, and the current results achieved. Rather than novel robot architectures or data processing algorithms, our focus here is on the challenges involved in deploying robots in the real world, and the corresponding lessons learned. We conclude the article with a discussion on our plans to integrate the sensors onto the APMs to achieve the vision of a specialty crop industry that is more efficient and more profitable, and therefore more competitive.

2 Reconfigurable mobility

Specialty crops growers have documented the need for nimble, capable and intelligent vehicles that can be flexibly tasked to perform various functions in orchards and nurseries [16]. Existing vehicles (e.g., mobile harvesting platforms, tractors) either have narrow functionality or are too expensive to be used sparingly. To attend this need, we are developing a family of Autonomous Prime Movers, or APMs, autonomous vehicles and platforms for a variety of orchard operations and plant science data collection.

Fig. 2 First autonomous prime mover, based on a Toro MDE eWorkman vehicle

2.1 APM hardware

The first vehicle in the APM family (henceforth denoted simply as APM) is a drive-by-wire electric vehicle based on the Toro MDE eWorkman (Fig. 2). Work on retrofitting the eWorkman included the installation of steering and brake motors, motor amplifiers, steering and wheel encoders, and emergency stop buttons; and weatherization of major components to allow them to withstand a typical orchard environment on a dry day or under light rain. Figure 3 presents the high-level block diagram of the drive-by-wire system. A microcontroller connects to an external control and navigation computer and to sensors via an Ethernet switch.

Upon completing the retrofitting of the vehicle, we installed an onboard computer and laser sensors for safe row following and turning. The computer is a ruggedized Dell laptop especially configured to withstand harsh environmental conditions. Two laser rangefinders were mounted in the front of the vehicle to provide a 240° field-of-view (FOV) in the direction of motion. The lasers were calibrated to yield range to objects in the world in a coordinate frame attached to the vehicle. A typical image returned by the lasers is shown on the left in Fig. 4.

2.2 APM software

The next step toward making the eWorkman an autonomous vehicle was the development of the software that processes the returns of the laser rangefinders and guides the vehicle safely around the orchard by detecting and following rows of trees. We built upon our previous work on algorithms to track paths accurately while staying safe by avoiding

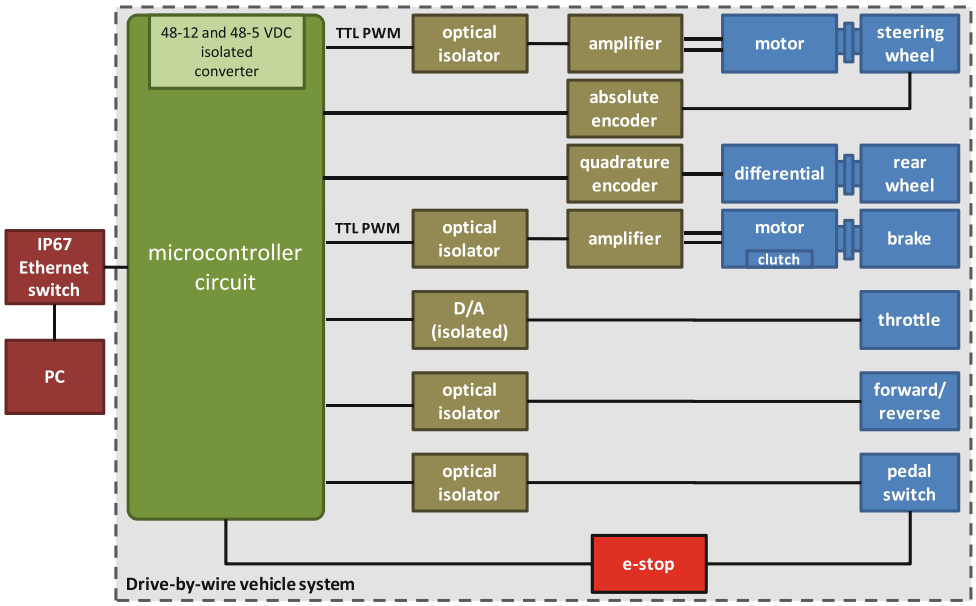

Fig. 3 Block diagram of the APM as a drive-by-wire vehicle system

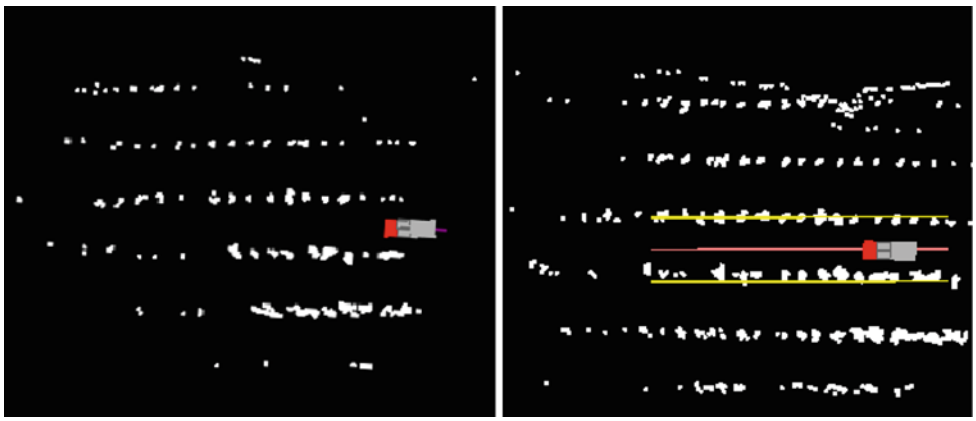

Fig. 4 *Left* Typical return of the laser rangefinders while the APM drives up an orchard row. The scan shown here represents the accumulated return of the lasers during 1 s. *Right* The row detection algorithm uses a Hough transform to find parallel lines in the obstacle map. These lines represent the left and right side of the row of trees

obstacles [4]; therefore, we focused on algorithms to reliably detect the edges of a row of trees and produce a path down the center for the vehicle to follow.

The row detection algorithm's main task is to determine the left and right edges of the row. To detect these edges, we use the Hough transform, a method commonly used to detect lines in images [6]. In our implementation, the Hough transform treats the map of objects detected by the lasers as the image in which to detect lines. Since the row detector needs to find both the left and right edges, it looks for the most likely pair of parallel lines from the Hough transform's results (Fig. 4, right). These detected parallel lines represent the rough line of the tree trunks.

Once the left and right side of the row are calculated, the center of the row is defined simply as the midway between the two lines. The system performs checks on the detected centerline to make sure it is safe for the vehicle to follow. If the line goes through obstacle points or if the vehicle is outside the detected row, the centerline solution is rejected. In this case, the system continues following the last detected row center.

The row detector was integrated into a vehicle guidance software module that can follow the center of the row, the edge of a row (using the Hough transform to look for only the right or only the left edges), and turn out of a row and into the next one. This module was combined with sensor and vehicle interfaces and a safety module to yield an autonomous software system which takes laser range data as input, finds the row of trees, and sends commands to the vehicle to drive down the row (Fig. 5).

To traverse orchard blocks safely and robustly, the APM needs an estimate of its current position—first, to determine when it is getting close to the end of a row, and should start looking for the end; and second, when the vehicle turns out of the row, to determine how far it has turned and when to stop turning. Although the APM is equipped with a high-precision GPS-assisted inertial navigation system, we use it only to generate ground-truth data for system testing and not to generate a position estimate. The rationale behind this choice is the cost of high-accuracy GPS systems, which could make the APM economically unattractive. Instead, we estimate vehicle position via dead reckoning, i.e., by processing

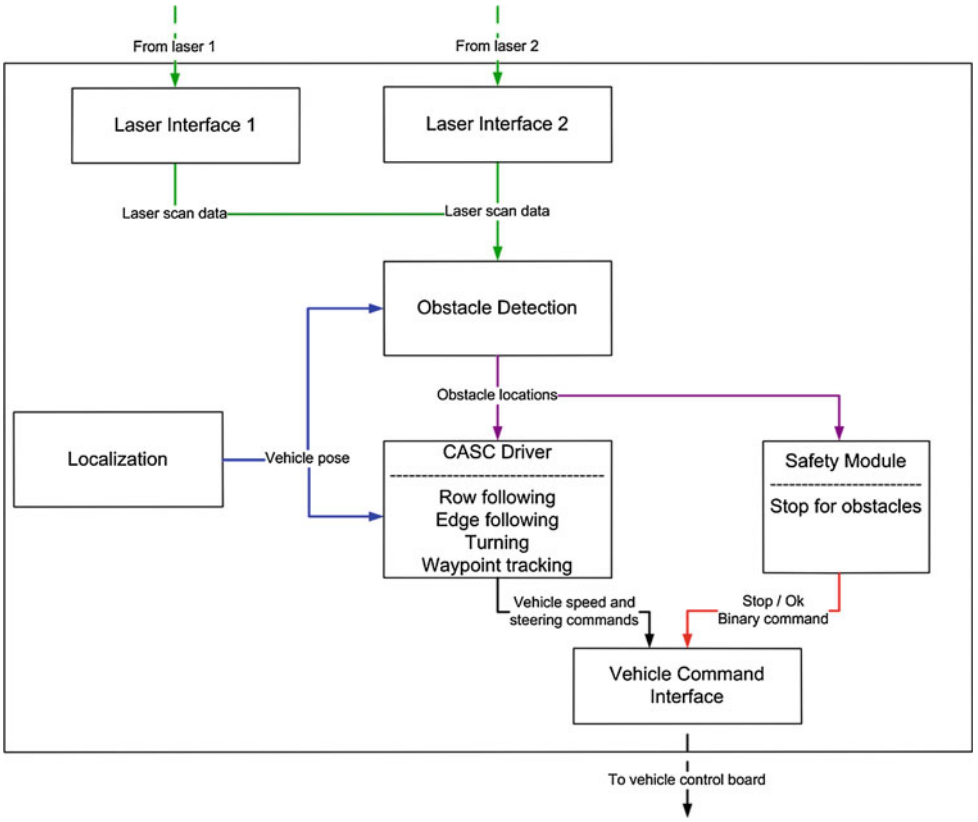

Fig. 5 Block diagram of the autonomous row following software system. Data come from the lasers, is converted into an obstacle map, and is then given to the driver module, which detects the row center and sends commands to the vehicle to drive

information from two encoders. One encoder is attached to the differential on the rear axle of the vehicle and measures how many revolutions the axle has completed. The other encoder is attached to the steering column and measures the angle of the wheels. The localization module (on the left in Fig. 5) combines these two measurements to determine how far the vehicle has traveled and in which direction, thus keeping a running estimate of the vehicle's position.

While the dead reckoning estimate is sufficient to let the navigation system operate, the accuracy of the estimate is poor. The autonomous navigation system must compensate for these deficiencies. The driver module must be able to detect the end of the row, since its estimate of the vehicle's position along the row could be incorrect. Likewise, when turning into a row, the module must detect the start of the row, since it cannot trust that the vehicle has turned correctly. In Sect. 4, we discuss how we intend to improve the performance of the localization subsystem using the tree lines and the tree trunks as landmarks.

2.3 APM testing

Orchard testing began in the spring of 2009, when the APM was completed. Initial tests were conducted at Soergel Orchards, in Wexford, PA, USA. The system performed well, routinely being able to follow a row, turn and enter the next row, and resume the row following behavior. Some unexpected challenges arose to proper row detection, though. The laser rangefinders were low enough to the ground that unmowed vegetation and uneven ground was sometimes detected and confused for tree canopy. This caused the autonomously detected row to veer slightly away from the actual row center (Fig. 6). At Soergel, with its 20 ft. row spacing, this was not a major issue.

In June 2009, we conducted a 1-week field trial at the Pennsylvania State University Fruit Research and Extension Center (FREC), in Biglerville, PA, USA. There, the much narrower, 12 ft. rows introduced two problems. The grass and ground detections continued to skew the calculated row center, and because the tolerance was much lower, the vehicle sometimes brushed tree branches. To mitigate the problem, we confined our tests in Biglerville to blocks with mostly level terrain.

Another place where narrow rows affected our success was in entering a row. The turn from one row to the next was executed by having the vehicle follow a prescripted path. But the rows were narrow enough that the slightest variance in row spacing was enough to cause the APM to miss the row entry, triggering the safety module to stop the vehicle (Fig. 7). We were able to achieve successful navigation of a block by tuning the parameters of the turn for each row. This field test illustrated that, in order to be generalizable and easy to use, the system needs to automatically detect the row to

Fig. 6 Sloped terrain (curving uphill) brings the ground itself into view of the lasers. This causes the row detector to produce a poor result. A low-pass filter on the row detection dampens this effect

Fig. 7 The APM turns into a row of trees at the FREC. At that time the turn command was based on a predefined row width. Any deviation from that width could lead to overshoot or undershoot, causing the safety module to stop the vehicle

be entered while turning and plan a path to the middle of the new row.

To demonstrate the capabilities of the APM to growers, we tested the robot carrying various pieces of farm equipment at the FREC. We hitched a mower to the back of the APM to have it mow the entire test block (Fig. 8, left). We also mounted NTech's WeedSeeker on its side to detect and selectively spray weeds growing beneath the trees (Fig. 8, right).

In July 2009, we tested the APM at Sunrise Orchards, a Washington State University-owned test planting in Rock Island, WA, USA, and Valley Fruit Orchards, a commercial operation near Royal City, WA, USA. We made two significant changes to the system between the Biglerville and Washington trips. First, we raised the height of the laser mounts on the vehicle, to prevent some of the spurious ground and vegetation detections. This made the row following more robust. Second, we implemented a new method for the autonomous software to detect and enter a row: rather than making a "blind turn" following a pre-defined path, the vehicle now actively searched for the next row and created the entry trajectory on-the-fly. The performance of the APM improved significantly with these changes. The vehicle was able to traverse entire orchard blocks at Sunrise Orchards with no tweaking of parameters for individual rows, using only the nominal row width and length and the number of rows to drive. The experiments at Sunrise culminated with a field day for local growers, in which we demonstrated the APM mowing an orchard block.

Further challenges to successful autonomous operation were presented by the dense tree canopies found at Valley Fruit. The trees at Valley Fruit were older and larger than those encountered in our previous tests, and the canopy width varied significantly along the row. This caused the detected row center to vary, and the vehicle drifted left and right as it traversed the row. Also, when attempting to enter a row, the width and density of the trees prevented the laser rangefinders from seeing into the row. Sometimes the system was successful, but frequently the autonomous system stopped the vehicle outside of a row, signaling that there was not

Fig. 8 *Left* The APM autonomously mows the grass in a block of trees. *Right* The APM autonomously sprays weeds with the NTech WeedSeeker

Table 1 Evolution of the performance of the APM between April and June, 2009 (partial list of all experiments conducted)

Date	Distance (km)	Rows	Row length (m)	Speed (m/s)	Experiment
05/08	3.2	18	179	1	Row following
05/15	2.2	12	179	1.5	Row following
05/15	3.6	20	179	1.5	Row following
05/20	11.6	62	179	1.5	Row following
05/29	4.4	25	179	2	Motor overheating assessment
06/12	6.3	35	179	1.5	Row following
06/19	11.6	65	179	1.5	Edge following
06/22	4.1	34	120	1.5	Row following
06/22	4.4	67	65.2	1.5	Edge following
06/23	11.8	181	65.2	1.5	Edge following
06/24	5.7	87	65.2	1.5	Mowing and spraying
06/25	6.7	103	65.2	1.5	Mowing demo at Penn State field day

Experiments on June 22–25 were conducted at the Penn State FREC in Biglerville, PA, USA. All others were conducted at Soergel Orchards in Wexford, PA, USA

enough data to find the row. All these issues are currently being addressed and results will be presented at the end of the 2010 summer field test season.

Our goal in Year 1 of the project (10/2008–09/2009) was to achieve 100 km of autonomous driving in real orchards. Table 1 presents the evolution of the performance of the autonomous system as we moved from single row following experiments in April 2009 to traversing full blocks by the end of June. Perhaps the most important milestones came on May 20th when the APM first broke the barrier of 10 km of uninterrupted autonomous driving; and on June 23rd when it traversed 181 rows of apple trees at FREC. In the 7 months since the APM was first tested as a drive-by-wire vehicle, it completed a total of 130 km of autonomous driving in various types of orchards: vertical ax (Soergel, FREC, Sunrise) and angled canopy and random fruiting wall (Valley Fruit); amidst young trees and fully developed canopies; in row spacings as small as 10 ft. and as large as 20 ft.; in orchards in Pennsylvania and Washington; and in test plantings and commercial orchards.

Fig. 9 The N. Blosi orchard agricultural platform converted into an automated orchard vehicle at the Pennsylvania State University Fruit Research and Extension Center

2.4 APM evolution

The second vehicle in the APM family is an orchard agricultural platform belonging to Pennsylvania State University (Fig. 9). The purpose of automating the platform is to make it a tool for augmented thinning, pruning, and harvesting, where workers can execute these activities without worrying with precisely driving the platform down the road at constant speed. The drive-by-wire design of the Toro eWorkman-based APM was mapped onto the N. Blosi such that the only changes are at the lowest level of autonomy control on the machine. Above this level, everything stays the same as on the first APM so that the autonomous row following system can be reused as is. The same microprocessor control board is used albeit with different software to account for the different inputs and outputs that are required because of the physical differences in the two vehicles. The N. Blosi platform was augmented with the same laser rangefinder units and computer presently installed in the APM, and emergency stop buttons that will put the hydraulic motor in neutral and apply the parking brake.

In recent tests at FREC the automated N. Blosi traversed 10 km of orchard rows over the course of 16 h. Further testing will be conducted in the fall of 2010 when the platform will be used for augmented harvesting trials.

Fig. 10 *Left to right* Original apple image before segmentation; binary image after segmentation; external and internal contours; candidate regions of IFW damage

3 Crop intelligence

While the APMs represent a breakthrough in orchard and nursery automation, where manually driven vehicles are the norm, they only serve a real purpose when carrying sensors, instruments, and implements that enable increased production efficiency and reduced labor costs. In this section, we describe a sample of our work in crop intelligence sensors, namely, those that automate insect infestation detection, crop load estimation, and tree caliper measurement. These and other application-oriented sensors are being or will be deployed on the APM to enable autonomous farm management practices.

3.1 Insect detection

Certain insect infestations can reduce crop yield and quality leading to significant economic loss. Currently, the only reliable way to detect insect damage—in particular, internal feeding worms, or IFW—in orchards is by human scouts performing visual inspection. Automated detection of fruit damage caused by IFWs, such as codling moth and oriental fruit moth, will reduce labor costs and allow for timely intervention thus mitigating further yield losses. We are developing computer vision and machine learning algorithms that can detect IFW-damaged regions on apple images. Once validated in the field, the algorithms can be run onboard the APM's computer to enable real-time, automated IFW detection.

3.1.1 IFW damage detection

The algorithm for detecting the evidence of damage by IFW within apples assumes that individual apples have been extracted from an image, such as done by the Scout crop load estimation system described later in this section. Given a segmented apple image, the algorithm classifies whether or not the apple has IFW damage. The algorithm consists of four steps (Fig. 10).

Step 1: A color-based segmentation algorithm is applied to the image using the learned color distribution of the target apples. The resulting binary image is smoothed out by applying erosion and dilation techniques.

Step 2: The external (blue) and internal (red) contours of the binary image are extracted using the chain-coding method [9].

Step 3: Candidate regions of IFW damage are detected via the following sub-steps:

(i) Keep the internal contours and discard the external ones.
(ii) Calculate the contour areas and keep those with area size between 40 and 500 pixels.[1]
(iii) Compute the smallest enclosing rectangle for each contour. Keep those whose ratio between the length of the longer side and the length of the shorter side is less than 2. The internal contours that result from these processing steps are labeled as candidate regions of IFW damage.

Step 4: Classify each candidate region as either "damaged" or "other." For this purpose, we developed a classification algorithm based on a support vector machine (SVM) [1]. The SVM algorithm is first trained using a large number of positive examples (i.e., IFW-damaged regions) and negative examples (i.e., other images that are not damaged by IFW). We used various image features to train the SVM algorithm, including the following:

(i) average RGB pixel values inside the contour region;
(ii) average RGB pixel values of the surrounding region of the contour;
(iii) the difference between the average RGB values inside the contour and the surrounding of the contour and the corresponding covariance values, and
(iv) texture features inside the contour region calculated via co-occurrence matrix analysis.

The four texture features analyzed in step (iv) were:

$$\text{Energy} : \sum_{i=0}^{255}\sum_{j=0}^{255} s(i,\, j)^2$$

[1] The values 40 and 500 were selected empirically based on the resolution of the image, the expected size of apples in the image, and the average size of IFW damage regions on apples.

$$\text{Entropy}: \sum_{i=0}^{255}\sum_{j=0}^{255} s(i,j)\cdot \log^{s(i,j)}$$

$$\text{Inertia}: \sum_{i=0}^{255}\sum_{j=0}^{255} (i-j)^2\cdot s(i,j)$$

$$\text{Local Homogeneity}: \sum_{i=0}^{255}\sum_{j=0}^{255} \frac{1}{1+(i-j)^2}\cdot s(i,j)$$

where $s(i, j)$ is a normalized entry of the co-occurrence matrix.

3.1.2 Experiments

Our initial experiments showed that the image features based on color [i.e., features (i), (ii), and (iii) from the list in step 4 above] perform the best. Figure 11 shows a typical example.

We tested the algorithm on 589 individual apple images. Out of these, 98 apples were damaged by IFW and the others were healthy, non-damaged apples. Table 2 shows the quantitative results obtained at the fruit level. At this level, the SVM implementation detects IFW damage in 91.8% of fruit actually bearing the injury, with a 13.2% false positive rate (i.e., flagging as damaged fruit that are actually healthy). Figure 12 shows some qualitative results.

Table 3 shows the quantitative results obtained at the IFW-damage level. At this level, the algorithm detects 90.3% of all damaged regions, but also flags as damaged a significant number of healthy ones. Clearly, reducing false positives is a priority we must continue to investigate.

Fig. 11 Examples of regions on the fruit classified as IFW-damaged

3.1.3 Image database

To test our IFW detection algorithm in real-world conditions, we collected a comprehensive image database between July 1st and September 30th, 2009. The database contains a total of 2,700 images of single apples. Three varieties are represented: Fuji, Golden Delicious, and York. Sample images for the Fuji are shown in Fig. 13. The following characteristics are noteworthy:

- the color of Fuji and York apples changes from green to red as they mature;
- the image backgrounds are complex and include leaves, branches, grass, sky, etc., complicating the image processing process. In particular, the green background makes the algorithm described in this section not applicable to the problem; and
- when apples become dark red, it is harder to distinguish IFW-damaged regions from healthy ones.

Since the images in the new database are significantly different than those taken during the winter of 2008–2009, the color-based segmentation algorithm did not work well. Therefore, we examined a new method that skips the segmentation step and directly applies the classification algorithm on every region of the image. We used the color pixel values in a 7×7 patch as the feature and the SVM to train and classify the data. Figure 14 shows some examples where all the IFW-damaged regions were correctly detected. Figure 15 shows examples where some of the background patches were detected as IFW-damaged regions (false positives). Extending this method is a topic of future work.

3.2 Crop load estimation

As part of the CASC project, Vision Robotics Corp. (VRC) is developing a crop load estimation system for medium- to high-density orchards. The system, the Scout, uses multiple stereo cameras on a vertical mast to scan fruit trees to determine the total crop load and the size and color distribution of apples in the block. The data can be output for any volume of space. In general, it is believed that growers will be interested in the data on a per-tree, per-row, or per-block basis, but it is also available for smaller samples—e.g., the top third of a tree. The Scout may collect data throughout the year, enabling

Table 2 Quantitative results of the IFW detection algorithm by fruit

Number of IFW damaged apples	Number of healthy apples	Number of correctly detected IFW damaged apples	Number of healthy apples that were falsely detected	IFW damaged apple detection accuracy	False alarm rate
98	491	90	65	91.8%	13.2%

Fig. 12 *Left* Original images of apples. *Right* The regions detected as IFW-damaged

Table 3 Quantitative results of the IFW detection algorithm by region

Number of IFW damaged regions	Number of correctly detected regions	Number of falsely detected regions		IFW damaged region detection accuracy
		Calyx	Others	
124	112	63	11	90.3%

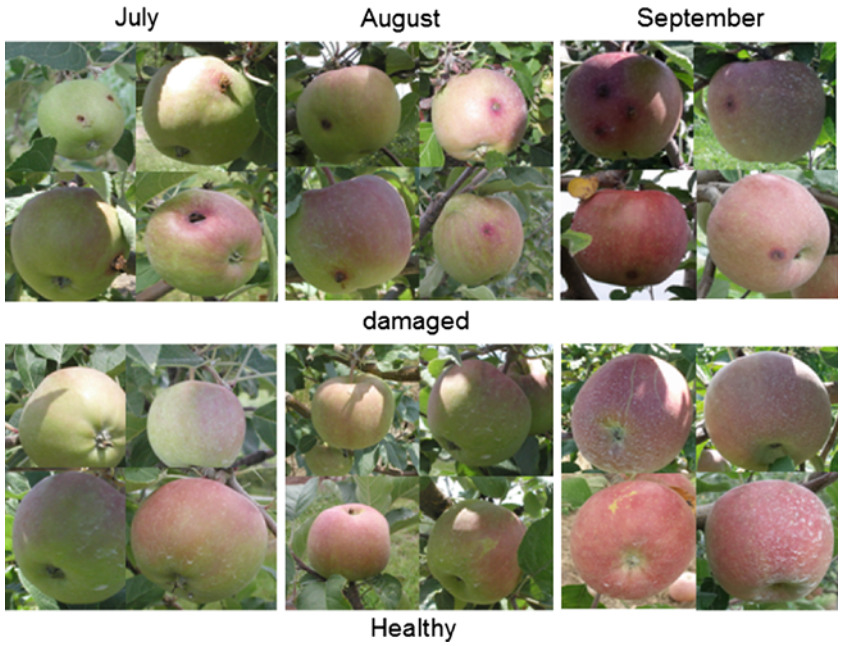

Fig. 13 Sample images of IFW-damaged (*top*) and healthy (*bottom*) Fuji apples taken during July–September, 2009

growers to better manage their crop during the growing season and better plan their harvest. Integrating the crop load estimate, GPS reference points and any additional data into a geographical information system (GIS) database creates a detailed yield map of the orchard for precision farming. The Scout's key specifications are:

- data geometry: crop load collected and disseminated for any cubic section of the block;
- yield accuracy: the average error in the load estimate is <5%; accuracy increases as the size of the section increases;
- sizing accuracy: the average error in the size distribution for any reasonably-sized section is <10% for apples larger than 2″ diameter. Accuracy increases as the size of the section increases;
- fruit color: any apple color;
- fruit size: 1″ diameter or larger (may operate 25% slower with fruit smaller than 2″ diameter);
- operational requirements: day and night operation; and
- operating speed: >2 mph when scouting fruit 2″ diameter or larger.

3.2.1 Hardware development

The most recent Scout system is shown in Fig. 16. The platform is a trailer that is easy to convert from four-wheeled (for scouting) to two-wheeled (for road towing). The camera mast can be moved in-and-out, tilted and pivoted. Currently,

Fig. 14 Examples of correctly detected IFW-damaged regions. *Top row* Input images. *Bottom row* Processed images. Regions deemed to be IFW-damaged are marked by *pink* spots (color figure online)

Fig. 15 Examples of false positive detections (non-IFW-damaged regions classified as damaged). *Top row* Input images. *Bottom row* Processed images. Regions deemed to be IFW-damaged are marked by *pink* spots (color figure online)

the system uses two data collection computers, eight stereo cameras and a flash-based lighting system; the electronics will ultimately also include a microcontroller for low-level system control, a second camera mast, and other sensors. The VRC-designed cameras have features enabling them to operate in a wide range of lighting condition including: the ability to quickly change between two saved exposure settings, flash control, synchronous capture across multiple camera pairs, stable mounting plate and lens holders, and low f-number lenses.

3.2.2 Software development

Significant effort has been devoted to the development of the Scout's software subsystems for apple detection. The detection algorithm is divided into two portions, namely the front-end which considers images individually and identifies potential apples, and the back-end which tracks detected fruit between images, including between frames from multiple cameras.

More specifically, the front-end software examines each image looking for pixels that are potentially parts of apples. This determination is based upon factors, such as color, texture, and shape. The goal of the front-end is to ensure that all potential apples are identified.

The back-end software tracks each potential apple identified by the front-end between images as the cameras are moved along the row and between the different camera pairs. By accurately knowing the apple position, the system determines whether apples seen from different perspectives or different cameras are the same or different. Because the views change significantly as the Scout moves, non-apple features identified by the front-end are only tracked through a small number of images. For example, a leaf or a knot on a tree

Fig. 16 The Scout in the field. The mast, which houses the cameras and flash units, is mounted on an arm that can be moved to adjust the mast position relative to the trees

trunk may look round from one viewpoint, but not from others because those features are not spherical. By amalgamating the information from multiple images, the back-end makes a final determination as to which of the potential apples are actual apples.

The detection software is assisted by a number of hardware-related features. The flash units allow the front-end portion of the software to more easily detect apples, particularly in the case of green apples. Another significant component is the use of software to control the cameras. When a flash is used with multiple cameras, it is essential that all cameras capture images at precisely the same moment; this synchronization and control is coordinated through the software. Also, the software implements multi-exposure through the notion of "contexts," where each context can utilize a different exposure setting. The system progresses through a sequence of contexts, enabling multi-exposure. During the first year of development, a desired average pixel intensity was pre-set for each context, and auto-exposure algorithms adjust the exposure levels to achieve the targets.

During 2009, the Scout was integrated with the APM for precise motion control. This coupling included the definition of an interface and set of protocols through which the APM instructs the Scout when to start and stop data capture and relays GPS data from the APM to the Scout. To facilitate testing, simulators for these control and GPS data interfaces were created.

Fig. 17 The Scout integrated with the APM at Valley Fruit Orchards, Royal City, WA, USA

3.2.3 Field tests

In 2009, VRC conducted two extensive sets of field tests. The first set of tests, at Valley Fruit Orchards near Royal City, WA, USA, was conducted in July (Fig. 17). Images from a section of the orchard with verified hand-counted data were collected repeatedly while modifying one system variable at a time from a base configuration. Comparing test results from each of these runs determined which variables improved performance. Because variables were only adjusted one at a time,

Fig. 18 Representative Jazz and Granny Smith apple trees scanned during field tests

analysis determined their relative effect. Working with the APM, VRC tested 17 configurations by varying the following parameters: ground speed, mast position relative to tree (distance and height), mast orientation relative to tree, camera angle relative to the mast, lighting, flash and multi-exposure, picture frequency along row, and Sun angle (including after sunset).

In the second field test period, which took place at the Allan Brothers' orchard in Othello, WA, USA, VRC further refined the hardware configuration and gathered data to validate the estimation software. Figure 18 shows representative pictures of the Jazz and Granny Smith trees scanned. The Jazz sections used vertical trellises and contained relatively sparsely growing apples in a thin canopy. Conversely, the Granny Smith sections used the angled-V configuration and contained fruit which were growing in tight clusters with a dense canopy.

To enable a significant analysis, a team from the Washington Tree Fruit Research Commission (WTFRC) hand-collected data from 100 ft. sections in each of two rows of Jazz and two rows of Granny Smith. Each 100 ft. section was divided into approximately 3 ft.-wide segments, and the number and size of apples contained within each segment was recorded.

The first aspect of the Scout system that we evaluated was the ability of the cameras to capture images of sufficient quality and including enough of the apples for accurate crop load estimation. VRC manually counted the apples within a 15 ft.-long section of each apple variety by reviewing pictures captured (from all camera pairs) as the Scout moved down the row. The Scout collected images in which a human could identify 91.53 and 98.77% of the actual fruit for the Jazz and Granny Smith sections, respectively. Therefore, for the test cases analyzed the vast majority of apples are visible in the pictures, demonstrating that there is sufficient information in the images for accurate crop load estimates. The remainder of the project involves implementing a software system with sufficient performance to produce accurate crop load estimates from these images.

To evaluate the performance of the front-end portion of the software, 33 images were selected at random from a scan of the Jazz section. For each image, a human identified the number of human-visible fruit that were correctly detected as possible candidate apples by the software, and the total number of human-visible apples. A representative image for this analysis is shown in Fig. 19. Here, squares denote regions of the image which were marked by the software as candidate portions of apples. One-hundred percent of the apples passed through the front-end of the detection software.

Fig. 19 Representative image showing regions identified as candidate apples in the individual image (*squares*) and regions decided to be apples after processing multiple images (*circles*)

To analyze the back-end portion of the software, the same 33 randomly selected images from the Jazz section were reviewed, after tracking over multiple images, to determine the correctly identified apples and the false positive detections. A representative image is given in Fig. 20. Here, solid circles represent apples identified by the back-end portion of the software. The total numbers of fruit, correctly detected fruit, and falsely detected fruit in the analyzed images are given in Table 4. Falsely detected fruit are primarily caused by misalignment between camera pairs and the fact that the background dirt is red in color; both issues are easily resolved.

The final analysis was to compare the load estimates produced by the software with the true values determined by the WTFRC team. The average relative error

$$\frac{\text{estimated yield} - \text{true yield}}{\text{true yield}} \times 100\%$$

in the crop load estimate over the entire 100' was 1.77% for the Jazz section. For the Granny Smith section, only the first five segments have currently been analyzed, with average relative error in crop load equal to −14.81%. It was observed by the WTFRC team that several apples were located on the border between two segments; thus, these apples may have been counted in different segments for the hand and Scout counts. As shown in Fig. 21, the estimates for Jazz were observed to follow quite closely the true count in each 3 ft. segment, suggesting a strong match to the true load distribution. Table 5 gives the software and hand-collected counts for the first five of the approximately 3 ft.-wide segments of one of the Granny Smith test sections. Here, the estimates tended to be lower than the true counts, in part due to the software missing individual fruit in the centers of large clusters. These observations help guide the planned improvements to the estimation algorithms. VRC anticipates developing enhancements to both the front- and back-end to improve load estimates. In particular, we believe that a statistical model that adjusts the software output to produce a final crop estimation can be developed. For example, such a model would compensate for orchards in which fruit is growing in dense clusters and thus may not all be visible to the cameras.

Fig. 20 Representative image showing apples identified by the (back-end) software

Table 4 Counts of correctly and falsely detected fruit for the Jazz section

Apple variety	Total apples	Number correctly detected in back-end	Number falsely detected in back-end
Jazz	169	155	27

3.3 Caliper measurement

Caliper, a measure of growth and marketability, is manually measured in tree crops, consuming time of a fallible and diminishing labor pool. Measuring caliper and counting trees is a costly process in which data are not spatially or temporally recorded; instead the information is hand-logged revealing little or no information about management practices and providing incomplete inventory projections. We are developing a fast and low-cost method to count trees, record geospatial location, and measure caliper in tree cropping systems with the aim to increase production efficiency, provide models of plant growth and assist in precision management.

Upchurch et al. [13] have described a system that uses an ultrasonic transducer for measuring tree trunk diameters. Diameters of circular objects were calculated using the time interval for sound waves to travel from the transducer to the object and back to the sensor. A V-shaped hook was used to fix the back of the tree trunk relative to the sensor and the distance between the transducer and object decreased as the diameter increased. Such a device requires careful positioning by hand before a measurement can be taken. Henning et al. [5] have described a method to use a laser scanner to estimate tree diameter.

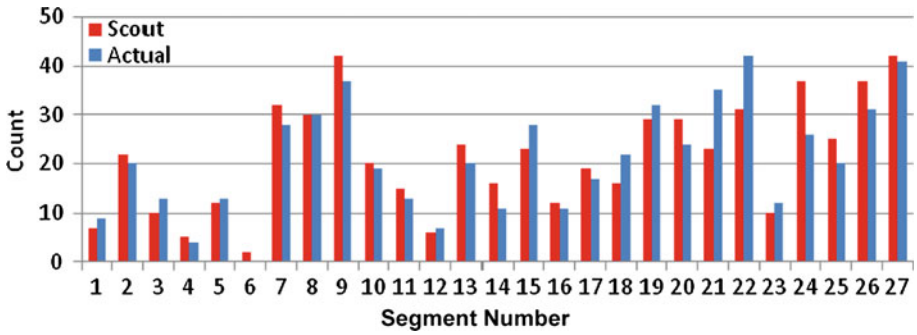

Fig. 21 Hand and software counts for 3 ft. segments of the Jazz block

Table 5 Hand and software counts for the first five 3 ft. segments of the Granny Smith block

	Segment 1	Segment 2	Segment 3	Segment 4	Segment 5
Actual count	27	37	36	39	23
Software count	23	29	36	31	19

Fig. 22 Rigid mount and camera (*inside circle*) on ATV used to collect imagery to assess the challenges to be addressed by the automated caliper and counter devices

Fig. 23 *Left* Caliper device tested at J. Frank Schmidt and Son Co. in Oregon. *Right* Device tested at Raemelton Farm in Frederick, MD, USA

Delwiche et al. [2] have described a system to count and size fruit and nut trees in commercial nurseries. An optical sensor was designed using a high-power infrared laser for illumination to allow operation with varying light conditions, including direct sunlight. The optical sensor was mounted on a cart and a rotary encoder was coupled to one of the wheels for displacement measurement. Signals from the optical sensor and rotary encoder were analyzed to determine trunk diameter, and running counts were maintained for the standard nursery size grades. Calibration tests showed that the system could measure trunk diameter with a standard deviation of 0.65 mm from a distance of 15–23 cm from the tree line.

Work on the caliper and tree counter started with visits to nurseries in Oregon and Washington where we interviewed producers to assess specific industry needs and to record images and video of field conditions. A camera mounted on an ATV (Fig. 22) was field-tested on staked shade trees of varying caliper and spacing. Video was taken on stems that were both sun-exposed and shaded. The speed of the ATV ranged from 2 to 4 mph.

The data collected were useful in identifying challenges that needed to be addressed by the automated caliper device:

- the speed of travel at which the device can collect and interpret data;
- the need to mount the device approximately 12″ above the ground to miss obstacles and in an angle to remove background interference from the adjacent row;
- the need to differentiate tree trunks from stakes; and
- the need to measure caliper in the presence of weeds in front of the tree trunk.

Based on these findings we designed and built a prototype caliper device. This device has been tested in a variety of field conditions in Oregon, Washington, Pennsylvania and Maryland (Fig. 23).

We use a relatively simple approach to estimating caliper. Two planes of laser light are projected onto the scene and imaged with a camera. Caliper is computed as the average of the line widths found in the corresponding image. Since it is not possible to guarantee a fixed distance between the sensor and tree, we calculate a scale factor S that relates pixels in the image to actual distance. S is computed in each image simply as the ratio p/v, where p is the distance between the two planes of light in the image (in pixels), and v is their actual physical distance (Fig. 24).

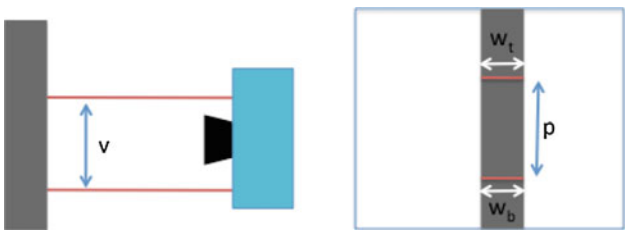

Fig. 24 Principle of operation of our caliper. *Left* Two parallel planes of light separated by a distance v are projected onto the tree trunk. A camera placed between the laser lines images the scene continually. *Right* A stylized image of the tree taken by the camera showing pixel quantities p, W_t and W_b

Fig. 25 Two examples of images of the two lines projected on a tree. The width of each line, in conjunction with the scale factor S, can be used to produce an estimate of the tree caliper. Here the distance of the camera to the tree is significantly different but the estimate of caliper varies little

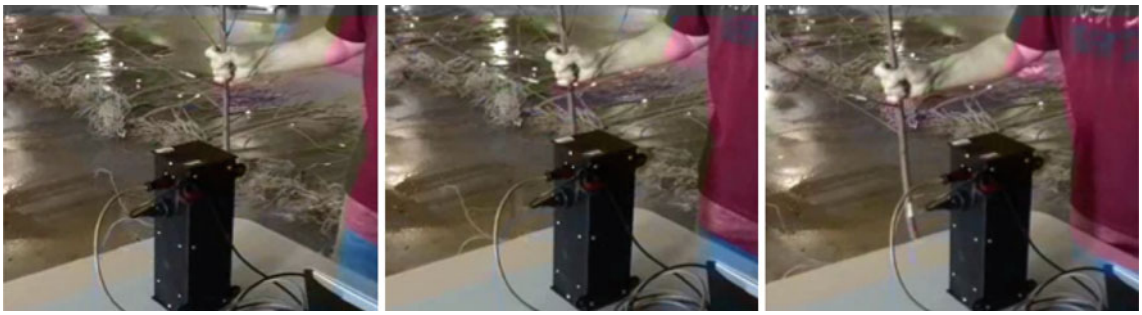

Fig. 26 For indoor testing the caliper device was placed on a table and individual trees were moved (*right to left*) in front of the caliper to replicate the relative motion of the device in the field. In our data set, we get between five and nine measurements per tree

The chief task of our computer vision algorithm is the robust computation of W_t and W_b, the widths in pixels of the projected lines on the tree in the corresponding images. While this is a simple computation in theory, robustly locating these lines is difficult given that the camera does not remain completely horizontal as it moves through the field and the presence of low branches and leaves produces extraneous reflections. Once S has been computed, caliper is then computed as the average of W_t/S and W_b/S (Fig. 25).

3.3.1 Results

Here, we report results from two controlled tests conducted in April and May 2010 at Eisler and Adams County nurseries in Pennsylvania. Tests at Adams County Nursery were conducted inside a storage warehouse without control of ambient lighting but in the absence of direct sunlight. Tests at Eisler Nurseries were conducted outdoors in bright daylight conditions.

3.3.1.1 Indoor testing In these tests, the caliper device was placed on a table and kept stationary for the duration of the experiment. Trees were individually moved in front of the laser as shown in Fig. 26.

This process was used to measure 120 trees twice. Caliper estimates were logged for later analysis. Each tree was also measured using a handheld digital caliper and were graded by nursery professionals into standard grades.

Figure 27 shows each data point logged by the system, including multiple estimates of individual tree caliper. Figure 28 shows the data after multiple readings of each tree have been median-filtered. Both figures show errors in estimation of caliper—this is the difference between caliper measurement using a hand caliper and the estimate from our device.

The data in this plot show that:

- the error seems to be relatively unchanged over trees that range from 7 to 25 mm;

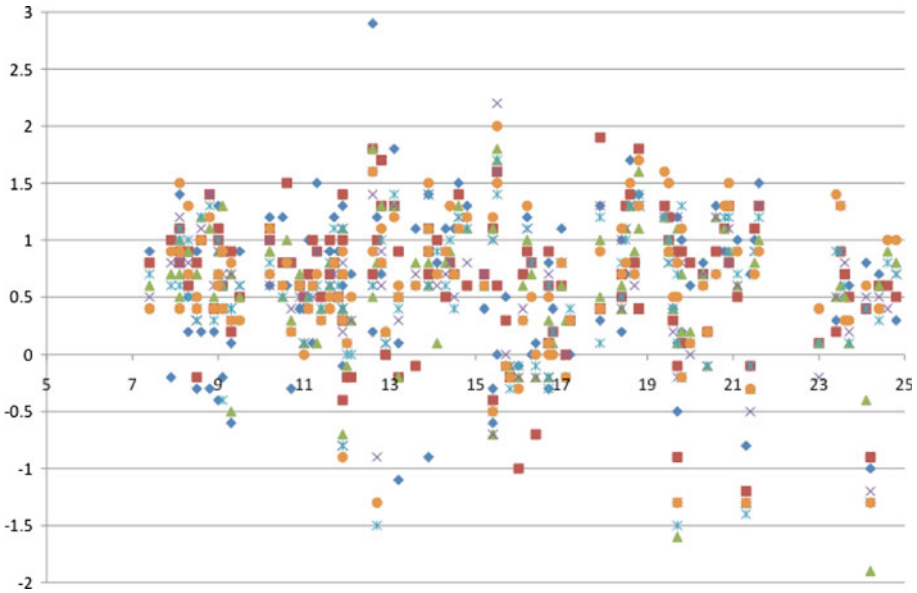

Fig. 27 Error (in mm) in caliper estimation vs. ground truth. This graph includes all caliper measurements made for each tree. The different *symbols* correspond to different readings taken for the same tree. Positive errors correspond to overestimates

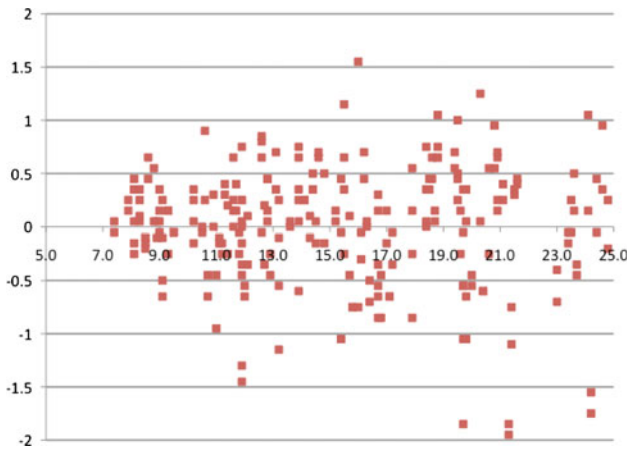

Fig. 28 Error (mm) in caliper estimation versus ground truth after filtering. The multiple measurements made for each tree were median-filtered to yield a single result per tree. Also, the bias (mean error) has been removed. The standard deviation in the error is now 0.56 mm

- there is a small (0.65 mm) bias (constant with tree size) of overestimating caliper. This is most likely due to "blooming" in the camera where the charge from saturated pixels spread into neighboring pixels;
- the standard deviation of the error is 0.68 mm.

Since our method delimits all the readings from a single tree, we can use a median filter to improve our estimates.

3.3.1.2 Outdoor testing For outdoor testing, we attached the caliper device to a small cart using a four-bar linkage that was kept at constant height using a rubber wheel. Figure 29 shows the caliper as it was deployed in the field. In one experiment, we calipered 50 trees while the caliper was moved at approximately 3 mph. We repeated this experiment five times to get a total of 250 caliper estimates.

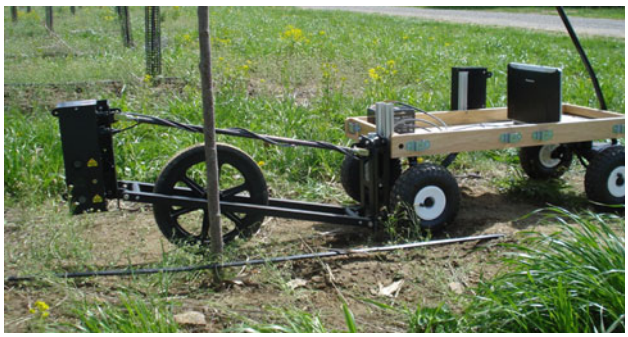

Fig. 29 For outdoor testing the caliper was mounted on a four-bar linkage and attached to a small cart that was towed through a field nursery

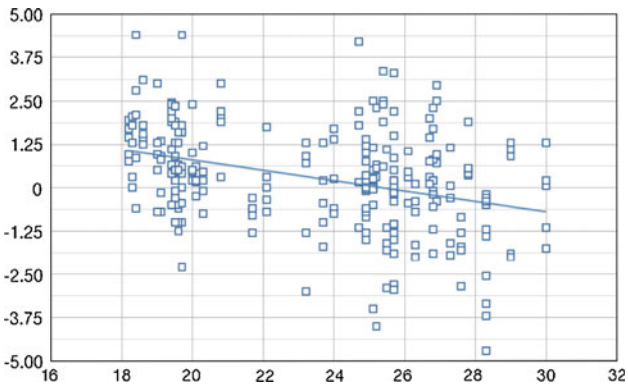

Fig. 30 Error (mm) in caliper estimates versus ground truth after median filtering. These data show a noticeable trend in the error as a function of the tree diameter. Without adjustment, the standard deviation is 3.2 mm

Figure 30 shows the error in caliper estimates after they have been median-filtered. At 3 mph, the number of hits on a tree is smaller (typically 2–3) and because the ground is not

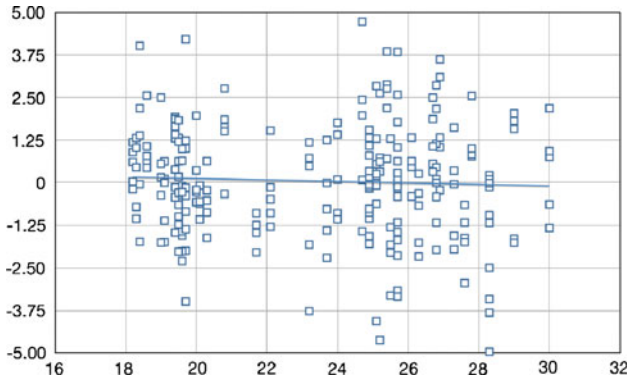

Fig. 31 Error in caliper estimates vs. ground truth after a trend and bias have been removed. Most estimates are now within ±2.5mm. The standard deviation is now 1.6 mm

even, it is not possible to keep the caliper pointing to the same place on the tree, so the data show a larger spread when compared to the indoor experiments. This chart shows a noticeable linear trend in the error. Since these trees have wider diameter, it is possible that the edges are not detected accurately. We can, however, adjust for this as it is a known bias.

Figure 31 shows the same data as above after the trend and bias have been removed. This processing is able to reduce the standard deviation by a factor of two.

Here we have reported first results with a device that can measure tree caliper "on-the-fly" in shade and fruit tree nurseries to automatically generate a database of tree caliper. Because the device is capable of determining when one tree leaves the field of view and the next one comes into view, it can also count trees. We have tested the device both indoors and in the field. Since our device uses infrared lasers, the significant differences between the indoor and field experiments have to do with ambient lighting conditions and with the ability to position the device to measure a tree trunk at the same position repeatedly. Generally, much less laser power is necessary when operating indoors and low-cost components can be used. Our experiments show that it is reasonable to expect an accuracy of approximately ±1 mm indoors and ±2.5 mm outdoors. It should be possible to get more accurate data by increasing the frame rate of the system (faster data acquisition) or conversely by slowing down the motion of the caliper. In the near future, we plan to test the device in large nurseries in Washington, Oregon and California and gather significantly larger datasets.

4 Conclusion

At the end of one year developing reconfigurable mobility and crop intelligence technologies for specialty crops we learned a significant number of lessons that will guide our work in the next three years of the project. Among the most important ones we list:

- Stakeholder involvement is fundamental to keep the project aligned with the needs of the specialty crops industry. By testing the APM and the crop intelligence sensors in commercial orchards and nurseries, and receiving *in loco* feedback from growers, we are able to continuously refocus our work on systems that are useful in the real world.
- Autonomous row entry using only laser data and vehicle odometry is much harder to execute reliably and repeatedly than autonomous row following. We were able to improve the robustness of row entry by incorporating row detection into the entry algorithm.
- The current odometry-based localization solution is sufficient to enable row following but not to enable georeferenced data collection. We intend to overcome this deficiency developing GPS-free, accurate localization algorithms using the tree rows and the trees themselves as landmarks (e.g., [7,12]) as well as landmark-free methods that build and use geometric maps of the environment [3].
- The APM and the applications it enables will only become a reality if it's equipped with a user interface designed with growers and farm employees in mind. To achieve this goal we recently initiated a formal interface design process based on methods and tools from the human-computer interaction area.
- The IFW damage detection algorithm can be complemented by deploying digital traps that automatically count insects that enter them. We created fifteen such traps by retrofitting commercial bucket traps with custom electronics and software and tested them in orchards in PA and WA. The traps are indeed capable of counting insects but capture fewer insects than regular traps. We believe this is due to the electromagnetic field created by the microprocessor in the digital traps. We are currently investigating how to overcome this problem.
- To continue improving the crop load estimation performance, the first step is to further quantify the existing detection and sizing performance. Concurrently, we will analyze the software to determine which algorithms require refinement and begin an optimization process to decrease the processing time. VRC will also evaluate and adapt the scouting platform with special attention to reliability in field conditions, including environmental temperature. The system will undergo field trials in production orchards during the summer and early fall of 2010. The crop load estimation results from these trials will be structured such that they can be input into a geographical information system as described next.
- The crop intelligence sensors are of little value to growers if the information resulting from the data collected

is not presented in a georeferenced, easy-to-manipulate graphical interface. We are developing a geographical information system capable of collating data collected by human scouts, fixed sensor networks, and mobile sensors mounted on the APM or other platforms such as tractors. This system publishes data in an OpenGIS standard format—currently KML—to be displayed by tools such as Google Earth. The interface currently allows growers to select and display data along spatial and temporal dimensions to infer crop and plant status relative to a variety of physical conditions (e.g., soil temperature, leaf wetness, etc.). Our goal is to make this a tool for advanced decision-making that automatically deploys the APM on targeted data collection missions or to execute specific operations.

Acknowledgments CASC is funded by the USDA SCRI program under award no. 2008-51180-04876. The authors would like to thank the owners and managers of the various orchards and nurseries cited in this paper for providing labor and land for our tests.

References

1. Bishop CM (2006) Pattern recognition and machine learning. Springer, Berlin
2. Delwiche M, Vorhees J (2003) Optoelectronic system for counting and sizing field-grown deciduous trees. Trans ASABE 46(3): 877–882
3. Fairfield N, Kantor G, Wettergreen D (2007) Real-time SLAM with octree evidence grids for exploration in underwater tunnels. J Field Robot 24(1):3–22
4. Hamner B, Singh S, Roth S, Takahashi T (2008) An efficient system for combined route traversal and collision avoidance. Autonom Robots 24(4):365–385
5. Henning JG, Radtke PJ (2006) Detailed stem measurements of standing trees from ground-based scanning lidar. For Sci 52(1): 67–80
6. Illingworth J, Kittler J (1988) A survey of the Hough transform. Comput Vis Graph Image Process 44(1):87–116
7. Leonard JJ, Durrant-Whyte HF (1991) Simultaneous map building and localization for an autonomous mobile robot. In: IEEE/RSJ international workshop on intelligent robots and systems, pp 1442–1447, November 1991
8. Pollack S, Perez A (2008) Fruit and tree nuts situation and outlook yearbook 2008. USDA Economic Research Service, p 29
9. Rosenfeld A, Kak AC (1982) Digital picture processing. Academic Press, New York
10. Singh S, Baugher T, Bergerman M, Grocholsky B, Harper J, Hoheisel G-A, Hull L, Jones V, Kantor G, Koselka H, Lewis K, Messner W, Ngugi H, Owen J, Park J, Seavert C (2009) Automation for specialty crops: a comprehensive strategy, current results, and future goals. Paper presented at the 4th IFAC international workshop on bio-robotics, information technology, and intelligent control for bioproduction systems, Champaign, IL, September 2009
11. Suarez L, Zarco-Tejada PJ, Sepulcre-Canto G, Perez-Priego O, Miler JR, Jimenez-Munoz JC, Sobrino J (2008) Assessing canopy PRI for water stress detection with diurnal airborne imagery. Remote Sens Environ 112:560–575
12. Tully S, Moon H, Kantor G, Choset H (2008) Iterated filters for bearing-only SLAM. In: IEEE international conference on robotics and automation, pp 1442–1448, May 2008
13. Upchurch BL, Anger WC, Vass G, Glenn DM (1992) Ultrasonic tree Caliper. Appl Eng Agricult 8(5):711–714
14. US Apple (2007) U.S. Apple Growers Could Lose $572.2 Million if the Farm Labor Supply Continues to Decline. http://www.usapple.org/industry/aglabor/econ_impact.pdf
15. USDA (2009) 2007 Census of Agriculture, United States, Summary and State Data, p 9
16. USDA (2007) Engineering solutions for specialty crop challenges, workshop report, Arlington, VA, April 2007. http://www.csrees.usda.gov/nea/ag_systems/pdfs/specialty_crops_engineering.pdf