Send your completed paper to Sandy Rutter at rutter@asabe.org by **April 30, 2010** to be included in the ASABE Online Technical Library.

Please have Word's AutoFormat features turned OFF and **do not include live hyperlinks.** For general information on writing style, please see http://www.asabe.org/pubs/authguide.html

This page is for online indexing purposes.

Author(s)

First Name	Middle Name	Surname	Role	Email
Jacqueline		Libby		jlibby@cmu.edu

Affiliation

Organization	Address	Country
Carnegie Mellon University, Robotics Institute	5000 Forbes Avenue Pittsburgh, PA 15213	

Author(s) – repeat Author and Affiliation boxes as needed--

First Name	Middle Name	Surname	Role	Email
George		Kantor		kantor@ri.cmu.edu

Affiliation

Organization	Address	Country
Carnegie Mellon University, Robotics Institute	5000 Forbes Avenue Pittsburgh, PA 15213	

Publication Information

Pub ID	Pub Date
1008605	2010 ASABE Annual Meeting Paper

2950 Niles Road, St. Joseph, MI 49085-9659, USA
269.429.0300 fax 269.429.3852 hq@asabe.org www.asabe.org

An ASABE Meeting Presentation

Paper Number: 1008605

Accurate GPS-free Positioning of Utility Vehicles for Specialty Agriculture

Jacqueline Libby

Carnegie Mellon University, Robotics Institute, 5000 Forbes Avenue, Pittsburgh, PA 15213, jlibby@cmu.edu

George Kantor

Carnegie Mellon University, Robotics Institute, 5000 Forbes Avenue, Pittsburgh, PA 15213, kantor@ri.cmu.edu

Written for presentation at the
2010 ASABE Annual International Meeting
Sponsored by ASABE
David L. Lawrence Convention Center
Pittsburgh, Pennsylvania
June 20 – June 23, 2010

Abstract. *This paper presents methods for determining the position of a robotic utility vehicle to sub-meter accuracy without the use of GPS. The approach we use is ideally suited for specialty agriculture applications such as orchards, where commercially available high-accuracy GPS systems are cost-prohibitive and GPS signal interference due to tree canopy can produce unreliable results. Solving the positioning problem provides a foundation for other tasks in precision agriculture that can be conducted with autonomous or partially-automated vehicles. Our algorithms use an Extended Kalman Filter with a suite of sensors. Given an initial estimate of vehicle position, sensors on the wheels and steering linkage are used to predict the path traveled, and then a scanning laser range finder is used to correct this predicted position by measuring the relative position between the vehicle and landmarks in the field. We have experimented with intentionally placed landmarks that use reflective tape, which can easily be identified with the laser. In this paper we present the motivation behind our techniques, the specifics of the algorithms we use, the experimental setups, and the results of field tests conducted during the summer of 2009 from apple orchards in Pennsylvania. Our results provide sub-meter accuracy, and suggest strong promise for reliable localization solutions for commercial applications.*

Keywords. Positioning, Precision Agriculture, Autonomous Navigation, Robot Analysis, Specialty Crops

1. Introduction

The ability to accurately determine the position of a moving vehicle is a fundamental component of many applications in precision agriculture. Knowing the location of the mobile platform at all times allows data collected by on-board sensors to be incorporated into maps that can be part of a GIS framework. By creating these maps, the vehicle can return to targeted areas and perform tasks such as spraying, thinning, and mowing. For instance, a variable-rate sprayer could be towed behind the vehicle and regulated based on the location in the field, thereby limiting the use of valuable resources such as water and pesticides. The role of positioning becomes even more critical in the case of automated or partially-automated vehicles that have the potential to reduce labor needs and improve productivity in specialty agriculture applications.

The Global Positioning System (GPS) is the *de facto* standard for solving the positioning problem, however it has many disadvantages for specialty crop settings. This is especially true in orchards where the line-of-sight to the GPS satellites is frequently occluded by trees and other structures. The occlusion problem does not occur in broad-acre crops, where GPS has been successfully used for many years (e.g., O'Conner et. al. 1996). Stentz et. al. (2002) used GPS in orange groves, however the tractors used in that work were tall enough that signal interference was not a large issue. Smaller vehicles, such as the electric utility vehicle used in this paper, must operate well below the tree line in apple orchards, as shown in figure 1 (left). Additionally, new fruit wall structures engineered to maximize light interception by the canopy (figure 1, right) further occlude the sky and rule out the possibility of mounting antennas above the canopy. Other outdoor environments with dense tree canopies, such as forests or almond groves, would pose similar problems.

Figure 1. Examples of environments that cause GPS signal interference. (Left) APM within a row of a dense orchard. (Right) A very young orchard with a v-structure fruit wall configuration.

Even without signal interference, GPS systems that provide sub-meter accuracy are prohibitively expensive for most specialty agriculture applications. Additionally, these systems do not provide any information about the orientation of the vehicle, which is necessary to accurately determine the position of objects observed by vehicle-mounted sensors.

This research is part of an effort in the nationwide CASC project (Comprehensive Automation for Specialty Agriculture) funded by the USDA to provide comprehensive solutions for specialty crops (Singh et. al. 2009). We are working with collaborators across the nation to tackle many angles of the problem, from robotics, to plant science, to agricultural economics, to education outreach. Our particular role at Carnegie Mellon University (CMU) is to automate a robotic utility vehicle, which can drive up and down orchard rows, performing a variety of tasks. The vehicle,

which we call the Autonomous Prime Mover (APM), is a drive-by-wire electric vehicle equipped with a variety of sensors to intelligently make sense of itself and the world around it.

Figure 2. The Autonomous Prime Mover, at Sunrise Orchards, WA. In view are two scanners in the front of the vehicle, a laptop on the passenger side, and two GPS antennas on the roll bar.

This paper describes efforts to date to estimate the pose (position and orientation) of the APM vehicle in real time without the use of GPS. Our approach uses sensors that are already on the vehicle for other purposes: wheel encoders for vehicle control, and laser range scanners for safety. Our algorithm can either replace or complement GPS to provide a more accurate and robust positioning solution. The algorithms are based on Bayesian estimation techniques for mobile robot localization and mapping (e.g., Thrun 2005). Wheel encoders provide dead reckoning, giving an initial estimate, and laser scanners are used to correct this estimate by determining relative position from landmarks in the environment. Such localization techniques have been successfully used in indoor settings, where sensors can easily detect sharp features such as the corner of a wall. (Grisettiyz et. al. 2005) Even in outdoor settings, man-made structures such as a window or the side of a building are often used. (Kummerle et. al. 2009) Guivant et. al. (2006) demonstrates Simultateous Localization And Mapping (SLAM) in a park-like setting with large, well-spaced trees. As shown in figure 1, the environments we are working in tend to have cluttered closely spaced trees, or trees with very small diameters, growing almost like a vine along a fruit wall. Since the use of natural features is quite challenging in this setting, we simplify the problem by installing reflective markers that can be used as landmarks.

This paper describes the APM platform in Section 2 and presents our algorithms for building maps and generating real-time position estimates in Section 3. Experiments used to validate these algorithms were conducted during the summer of 2009. These experiments and their results, which successfully demonstrate sub-meter accuracy, are described in Section 4. The paper concludes with a brief discussion of our plans for improving this work in the future.

Safety Emphasis

Working with robotic vehicles poses significant safety risks, though we have learned to mitigate them through experience and careful experiment design. The APM is equipped with three "E-

stops", large red buttons located at convenient positions on the vehicle that, which when pushed, shut off drive power and apply the brake. The vehicle is always operated with a human in the driver's seat, even when it is running in autonomous mode. However the most important safety precaution taken when field testing robots is to create and follow a comprehensive and rigorous test plan. This includes detailed check-lists that are followed when starting up and shutting down the robot.

On the application side, our work has the potential to increase safety for workers and managers operating in orchard environments. Robots are a natural fit for replacing humans in situations that contain environmental hazards, such as exposure to dangerous chemicals during spraying operations. Additionally, our technologies will reduce worker fatigue and the likelihood of repetitive stress injuries that result from monotonous tasks.

2. Experimental Platform

The APM shown above in figure 2 is a drive-by-wire electric vehicle, equipped with brake and steering motors that allow the vehicle to be controlled either by a human operator or by autonomous commands from an on-board computer. The vehicle is also equipped with various internal and external sensors that we use to estimate the position of the vehicle without GPS.

The internal sensors are used to measure the internal state of the robot. In this case, these are the encoders that measure distance traveled and steering angle. Encoder readings are received at a rate of 250 Hz. External sensors are used to sense the environment around the vehicle. In this case, these are two Sick LMS 291 laser scanners at the front left and front right sides of the vehicle, as shown in figure 2. The scanners send out a horizontal fan of these beams covering a 180° semicircle in 1° intervals, for a total of 181 beams in each scan. Each beam in the scan effectively measures the range (r) and bearing (ϕ) to the nearest object in its path. Figure 3 shows a top down view of the vehicle with the two laser scanners. Each scanner is rigidly attached to the vehicle and oriented 30° outward from the forward direction. The semicircles projecting from each laser are illustrating the 180° fan of emitted laser beams. Each laser sends a complete scan 35 times per second.

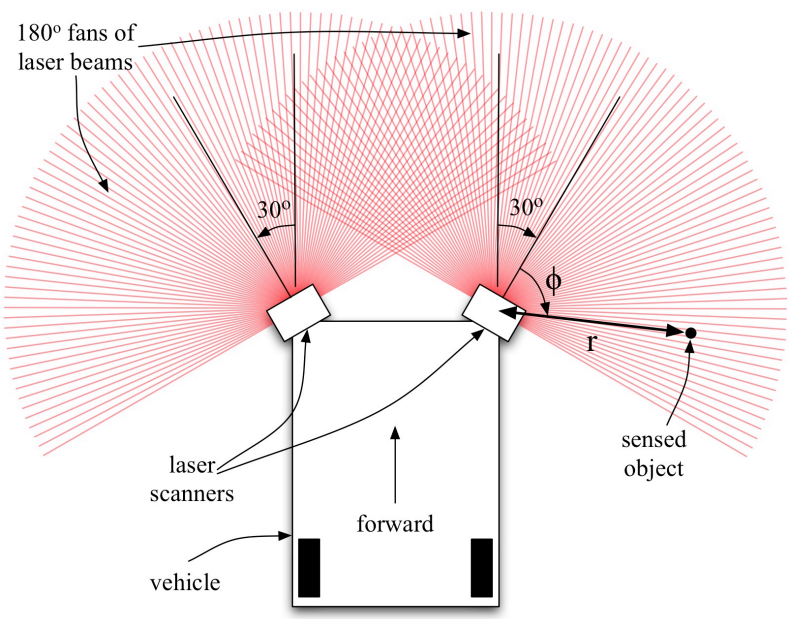

Figure 3. Top down view of vehicle and two laser scanners, showing fan of laser beams.

In addition to range and bearing, each laser beam returns an intensity value that indicates the reflectance of the corresponding object. Specifically, objects with retroreflective tape can be easily identified from the intensity value. In this paper, we use retroreflective objects placed throughout an orchard (Figure 4) as navigation landmarks.

Figure 4. Reflective tape placed in the orchard. (Left) Affixed to temporarily placed traffic cone. (Right) Affixed to a post already in orchard.

The vehicle is also equipped with a ruggedized Dell laptop, which receives measurements from the various sensors via Ethernet and USB connections.

An Applanix POS 220 LV high-accuracy positioning system also onboard the vehicle is used to provide ground truth to evaluate the performance of our localization algorithms. The Applanix fuses RTK corrected GPS, IMU, and distance-traveled encoder measurements to provide a 6 degree-of-freedom pose estimate. The position estimate (x, y, z) is typically accurate to within a few centimeters, and the orientation estimate $(roll, pitch, heading)$ is accurate to within 0.05 degrees. Applanix measurements are received at a rate of 100 Hz.

3. Landmark-based Localization Algorithm

In our approach, we combine vehicle motion measurements (from the distance-traveled and steering encoders) with range and bearing measurements to reflective landmarks at known locations in order to estimate the pose of the vehicle, which includes position (x, y) and heading (θ). This requires that a map describing the positions of all of the landmarks is known a priori. In this work, we create the map offline with the help of ground truth from the Applanix. Once the map is created, it is used in a localization filter that generates real time pose estimates.

3.1 Creating the Map

The map is a list of the (x, y) positions of the landmarks in the environment. The mapping step involves preprocessing the laser data to determine these positions. The landmarks are the reflective markers placed throughout in the environment, which the lasers read as high reflectivity returns.

The first step in creating the map is to drive through the orchard, passing each landmark multiple times from different viewpoints, recording data from both lasers and the Applanix. The

raw data is asynchronous, meaning that data from the various devices comes in with timestamps that do not match up. To deal with this, we interpolate the Applanix data in a post-processing step so that it matches laser log timestamps. This allows us to register a laser scan at each timestamp with a corresponding ground-truth vehicle pose coming from the Applanix.

Considering one laser return from one laser scan registered to a vehicle pose, we can project this into a world map. The laser return consists of a range, r, and a bearing, ϕ, which can be written in global coordinates as the point $\left(x_p^w, y_p^w\right)$ using the following equations:

$$\begin{bmatrix} x_p^w \\ y_p^w \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\left(\theta_v^w\right) & -\sin\left(\theta_v^w\right) & x_v^w \\ \sin\left(\theta_v^w\right) & \cos\left(\theta_v^w\right) & y_v^w \\ 0 & 0 & 1 \end{bmatrix}}_{H_v^w} \underbrace{\begin{bmatrix} \cos\left(\theta_l^v\right) & -\sin\left(\theta_l^v\right) & x_l^v \\ \sin\left(\theta_l^v\right) & \cos\left(\theta_l^v\right) & y_l^v \\ 0 & 0 & 1 \end{bmatrix}}_{H_l^v} \begin{bmatrix} r\cos(\phi) \\ r\sin(\phi) \\ 1 \end{bmatrix} \qquad (1)$$

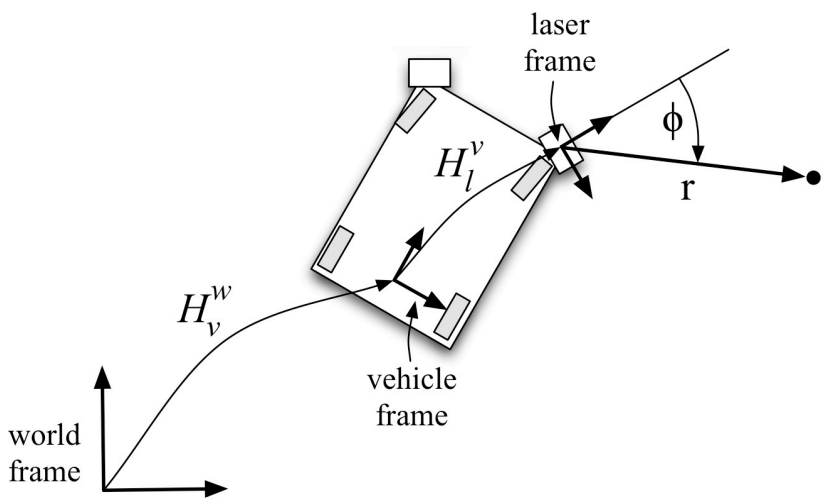

Figure 5. Illustration of the relationship between the laser frame, vehicle frame and world frame

$q_l^v = \left(x_l^v, y_l^v, \theta_l^v\right)$ is the pose of the laser with respect to the vehicle, which is a coordinate frame rigidly attached to the vehicle. $q_v^w = \left(x_v^w, y_v^w, \theta_v^w\right)$ is the pose of the vehicle with respect to the global coordinate frame. These poses are represented in equation (1) as homogeneous transformation matrices. (See Spong 2009 for more details.)

Figure 6 shows all the laser returns with high reflectivity projected into the world frame collected over the course of the mapping run. We throw out the low reflectivity returns, since they are not important for designating the landmarks in the map.

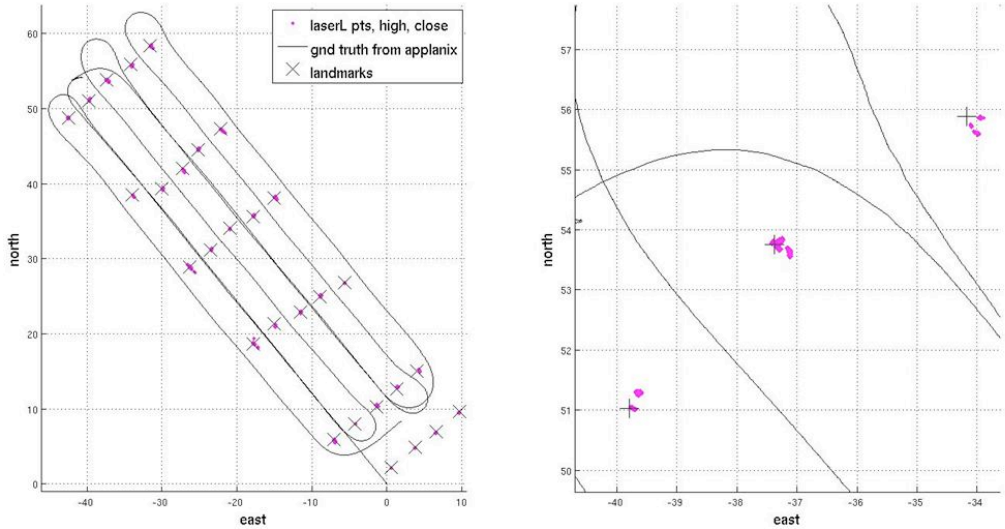

Figure 6. (Left) Pink dots are the projection of high reflectivity laser point returns onto a global coordinate plot. Crosses are the landmarks resulting from taking the means of the clusters. Black curve represents the vehicle trajectory during data collection. (Right) Zoomed in view to show a cluster of points, with the resulting landmark.

One can see from the figure that high reflectivity returns are clumped together, designating the positions of the reflective markers. The second step of the mapping is to cluster the resulting points. The average (x, y) value for each cluster of points is added to the list of landmarks in the map.

3.2 Motion Model

We will now discuss how the robot determines its position based only on its internal motion sensors. The term "dead reckoning" refers to position estimation without external references. In this case the external references are the landmarks in the map, which are recognized by the lasers (the external sensors). Putting the map and the lasers aside, we will first develop a motion model for the vehicle based only on its internal sensors, the steering and wheel encoders.

The principle behind the motion model is that given a vehicle pose at time t and the steering and wheel encoder values at time $t+T$, we can find the vehicle pose at time $t+T$. Assuming an Ackerman steering model for the differential drive four-wheel vehicle, we can simplify to a two-wheel bicycle model, as shown in figure 7.

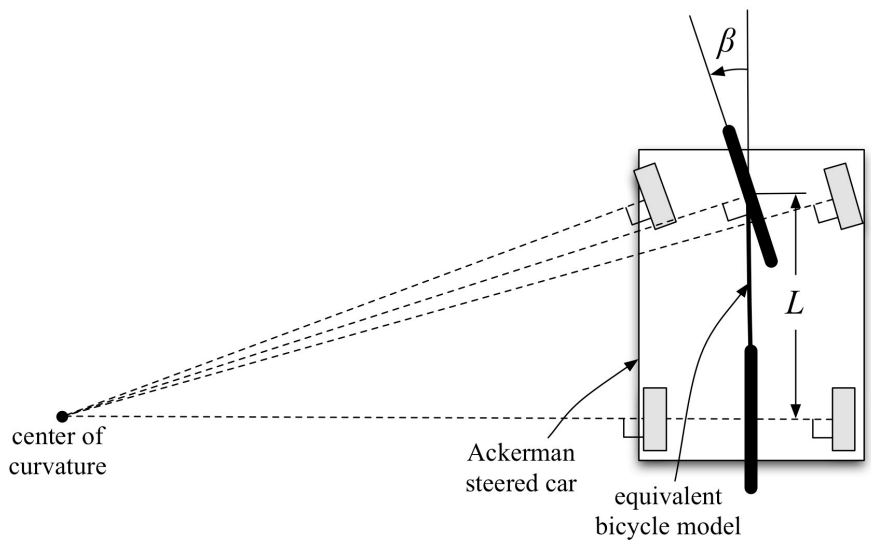

Figure 7. Differential drive and Ackerman steering model, simplified to a bicycle model.

An encoder mounted on the motor gives us an estimate of the forward distance traveled, d, as follows. At each time step, the encoder sends an integer representing the number of "ticks" (encoder increments) that have accumulated since the encoder was turned on. These accumulated encoder ticks measured at time *t* are denoted by $W(t)$. We convert these measurements to accumulated distance traveled at the center of the rear axle (which corresponds to the position of the rear wheel in the bicycle model) by converting ticks to meters,

$$D(t) = c_1 W(t) \qquad (2)$$

where c_1 is a constant scale factor which gives us the fraction of a meter travelled by the wheel for 1 encoder tick. The incremental distance traveled, represented by d, is then just the difference between the accumulated distance traveled at successive time steps:

$$d(t+T) = D(t+T) - D(t) \qquad (3)$$

The steering encoder is an absolute encoder that gives an integer representing the steering angle, denoted by $W_S(t)$. From this raw integer reading, we compute the steering angle, β:

$$\beta(t) = c_2 (W_S(t) - K_S) \qquad (4)$$

where K_S is a constant encoder offset value that designates what encoder tick reading corresponds to the steering facing straight and c_2 is a constant scale factor which gives us the fraction of radians for 1 encoder tick.

The forward velocity, v_k, and angular velocity, ω_k, are then computed with the bicycle model,

$$\begin{aligned} v(t+T) &= \frac{d(t+T)}{T-t} \\ \omega(t+T) &= \frac{\tan\beta}{L} v(t+T) \end{aligned} \qquad (5)$$

where L equals the length of the bicycle. We can now use these velocities to determine the relative pose at the next time step. Because the time step is small, we can make an Euler

approximation and replace the curved path shown in the bicycle figure with a point and shoot model,

$$q_v^w(t+T) = \begin{bmatrix} x_v^w(t+T) \\ y_v^w(t+T) \\ \theta_v^w(t+T) \end{bmatrix} = \begin{bmatrix} x_v^w(t) + Tv(t+T)\cos(\theta_v^w(t)) \\ y_v^w(t) + Tv(t+T)\sin(\theta_v^w(t)) \\ \theta_v^w(t) + T\omega(t+T) \end{bmatrix} = f\left(q_v^w(t), u_v^w(t+T)\right) \quad , \qquad (6)$$

where $u = (v, \omega)$ are the input velocities. (See Murray et. al. for more details.) We introduce a noise term, a, which accounts for noise in the sensors, wheel slip, and errors from the Euler approximation. We then rewrite equation (6) as:

$$q(t+T) = f(q(t), u(t+T)) + a(t) \qquad (7)$$

We model a as a zero-mean white Gaussian noise with covariance matrix C_a. Note here that $q = q_v^w$, and for the rest of this paper if we drop the extra notation we are assuming q takes this default value.

3.3 Measurement Model

As we repeat the dead reckoning step discussed above, errors accumulated at each time step will lead to a poor position estimate. We then use measurements from our lasers to the landmarks to correct for this error. We have our actual measurement to our landmark given by the laser, but we can also use simple trigonometry to model this measurement (range and bearing) based on the vehicle pose, the laser pose with respect to the vehicle, and a landmark location:

$$\begin{bmatrix} r \\ \phi \end{bmatrix} = h(q_l^w) = \begin{bmatrix} \sqrt{(x_l^w - x_b^w)^2 + (y_l^w - y_b^w)^2} \\ a\tan 2(y_l^w - y_b^w, x_l^w - x_b^w) - \theta_l^w \end{bmatrix} = \begin{bmatrix} h_1(x_l^w, y_l^w) \\ h_2(x_l^w, y_l^w, \theta_l^w) \end{bmatrix} \qquad (8)$$

(x_b^w, y_b^w) is the position of the beacon (or landmark) as listed in the map. Using the relationship $H_l^w = H_v^w H_l^v$, we can define q_l^w as a function q_v^w, along with the constant q_l^v:

$$q_l^w = \begin{bmatrix} x_l^w \\ y_l^w \\ \theta_l^w \end{bmatrix} = \begin{bmatrix} \cos(\theta_v^w)x_l^v - \sin(\theta_v^w)y_l^v + x_v^w \\ \sin(\theta_v^w)x_l^v + \cos(\theta_v^w)y_l^v + y_v^w \\ \theta_v^w + \theta_l^v \end{bmatrix} = \begin{bmatrix} g_1(x_v^w, \theta_v^w) \\ g_2(y_v^w, \theta_v^w) \\ g_3(\theta_v^w) \end{bmatrix} = g(q_v^w) \qquad (9)$$

We then rewrite equation (8) as:

$$h(q_l^w) = h(g(q_v^w)) = h(q_v^w) = \begin{bmatrix} h_1(x_l^w, y_l^w) \\ h_2(x_l^w, y_l^w, \theta_l^w) \end{bmatrix} = \begin{bmatrix} h_1(g_1(x_v^w, \theta_v^w), g_2(y_v^w, \theta_v^w)) \\ h_2(g_1(x_v^w, \theta_v^w), g_2(y_v^w, \theta_v^w), g_3(\theta_v^w)) \end{bmatrix} \qquad (10)$$

The function, $h(q_v^w)$, is our measurement model. It takes as an input the vehicle pose (along with the constant position of the associated landmark and the constant pose of the laser in the vehicle frame) and returns what should be the range and bearing measurement to that landmark. We introduce a noise term, b, which accounts for noise on the sensor readings, which is modeled as a zero-mean white Gaussian noise with covariance matrix C_b:

$$y(t) = h(q(t)) + b(t) \qquad (11)$$

3.4 Extended Kalman Filter

The extended Kalman filter (EKF) estimates the state of a system, and the uncertainty of that estimate. In this case, the state is the pose of the vehicle and the estimate is represented by the vector $\hat{q}$. The uncertainty is represented by a 3x3 covariance matrix P, which is the covariance of the estimate errors on our three pose variables. The EKF has two basic steps. The first is the prediction step that is done whenever a motion measurement is received, and the second is the update step that is done whenever a landmark measurement is received. The prediction step is based on the motion model, and gives a "prediction" of the state based on dead reckoning. The update step is based on the measurement model, and "updates" or "corrects" the prediction using a measurement to a landmark. The equations for the prediction step are:

$$\hat{q}(t+T) = f(\hat{q}(t), u(t+T)) \qquad (12)$$

$$P(t+T) = FP(t)F^T + WC_aW^T \qquad (13)$$

Note that equation (12) is the same as the motion model in equation (7), except that we have dropped the noise term and put a "hat" over the vehicle pose, q. Uncertainty added to the estimate due to the motion model noise term, a, is incorporated in the prediction using $2x2$ covariance matrix, C_a, on the input velocities. F and W are the Jacobians of the function, f:

$$F = \frac{\partial f}{\partial x} = \begin{bmatrix} 1 & 0 & -v(t+T)T\sin(\hat{\theta}(t)) \\ 0 & 1 & v(t+T)T\cos(\hat{\theta}(t)) \\ 0 & 0 & 1 \end{bmatrix} \qquad (14)$$

$$W = \frac{\partial f}{\partial u} = \begin{bmatrix} T\cos(\hat{\theta}(t)) & 0 \\ T\sin(\hat{\theta}(t)) & 0 \\ 0 & T \end{bmatrix} \qquad (15)$$

The prediction step happens whenever new wheel encoder values are received. The update step happens when the lasers detect a landmark by a high intensity return. When a high intensity return is received, the measurement is recorded as $z = (r, \phi)$. The algorithm must first figure out which landmark has been detected, if any, out of the list of landmarks in the map. The algorithm loops through all the landmarks in the list, and uses the measurement model, $(\hat{r}, \hat{\phi}) = h(\hat{q}_v^w(t))$ from equations (8), (9) and (10) to determine what the measurement would be to each landmark if the current estimate, $\hat{q}(t)$, was correct. It chooses the landmark that returns the closest $(\hat{r}, \hat{\phi})$ to the real measurement (r, ϕ). By "closest", we are implying a distance metric, and here we use the Mahalanobis distance which can be thought of as the distance in the space of sensor outputs, weighted by uncertainty. (Choset et. al. 2005) Figure 8 shows both the real and estimated measurements to an example landmark:

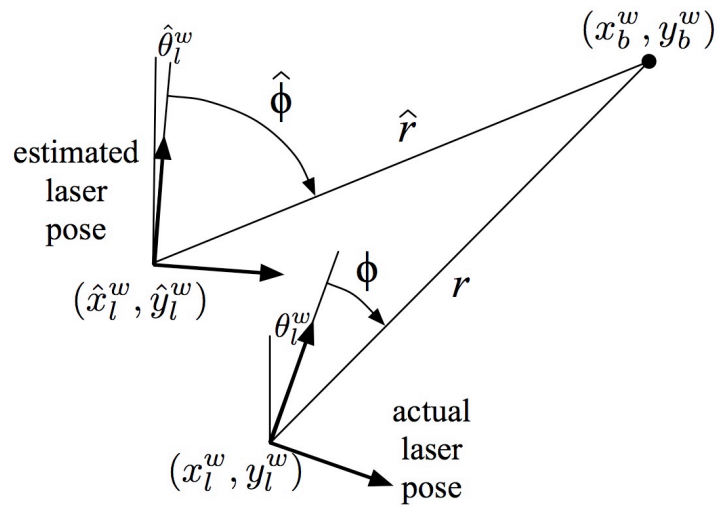

Figure 8. Determining error in pose estimate from a measurement to a landmark.

$\left(\hat{x}_l^w, \hat{y}_l^w, \hat{\theta}_l^w\right)$ is the estimated pose of the laser on the vehicle, whereas $\left(x_l^w, y_l^w, \theta_l^w\right)$ is the actual pose of the laser on the vehicle, which we do not know. With both the actual and estimated measurements, the robot can infer its error. We call the difference between these two measurements the innovation, η:

$$\eta = z - h(\hat{q}_k) = \begin{bmatrix} r \\ \phi \end{bmatrix} - \begin{bmatrix} \hat{r} \\ \hat{\phi} \end{bmatrix} \qquad (16)$$

We now write the equations for the update step:

$$K = P(t)H^T\left[HP(t)H^T + C_b\right]^{-1} \qquad (17)$$

$$\hat{q}(t) = \hat{q}(t) + K\left(z(t) - h(\hat{q}(t))\right) \qquad (18)$$

$$P(t) = (I - KH)P(t) \qquad (19)$$

Using $h\left(\hat{q}_v^w\right)$ as defined in equations (8), (9), and (10), we define the Jacobian, H, as:

$$H = \frac{\partial h}{\partial q_v^w} = \begin{bmatrix} \frac{\partial h_1}{\partial x_v^w} & \frac{\partial h_1}{\partial y_v^w} & \frac{\partial h_1}{\partial \theta_v^w} \\ \frac{\partial h_2}{\partial x_v^w} & \frac{\partial h_2}{\partial y_v^w} & \frac{\partial h_2}{\partial \theta_v^w} \end{bmatrix} = \begin{bmatrix} \frac{\partial h_1}{\partial g_1}\frac{\partial g_1}{\partial x_v^w} & \frac{\partial h_1}{\partial g_2}\frac{\partial g_2}{\partial y_v^w} & \frac{\partial h_1}{\partial g_1}\frac{\partial g_1}{\partial \theta_v^w} + \frac{\partial h_1}{\partial g_2}\frac{\partial g_2}{\partial \theta_v^w} \\ \frac{\partial h_2}{\partial g_1}\frac{\partial g_1}{\partial x_v^w} & \frac{\partial h_2}{\partial g_2}\frac{\partial g_2}{\partial y_v^w} & \frac{\partial h_2}{\partial g_1}\frac{\partial g_1}{\partial \theta_v^w} + \frac{\partial h_2}{\partial g_2}\frac{\partial g_2}{\partial \theta_v^w} + \frac{\partial h_2}{\partial g_3}\frac{\partial g_3}{\partial \theta_v^w} \end{bmatrix} \qquad (20)$$

Where, $\frac{\partial g_1}{\partial x_v^w} = 1$, $\frac{\partial g_1}{\partial y_v^w} = 0$, $\frac{\partial g_1}{\partial \theta_v^w} = -\sin\left(\theta_v^w\right)x_l^v - \cos\left(\theta_v^w\right)y_l^v$, $\frac{\partial g_2}{\partial x_v^w} = 0$, $\frac{\partial g_2}{\partial y_v^w} = 1$,

$\frac{\partial g_2}{\partial \theta_v^w} = \cos\left(\theta_v^w\right)x_l^v - \sin\left(\theta_v^w\right)y_l^v$, $\frac{\partial g_3}{\partial x_v^w} = 0$, $\frac{\partial g_3}{\partial y_v^w} = 0$, $\frac{\partial g_3}{\partial \theta_v^w} = 1$, $\frac{\partial h_1}{\partial g_1} = \frac{\left(g_1 - x_b^w\right)}{\sqrt{\left(g_1 - x_b^w\right)^2 + \left(g_2 - y_b^w\right)^2}}$,

$$\frac{\partial h_1}{\partial g_2} = \frac{\left(g_2 - y_b^w\right)}{\sqrt{\left(g_1 - x_b^w\right)^2 + \left(g_2 - y_b^w\right)^2}}, \quad \frac{\partial h_2}{\partial g_1} = \frac{-\left(g_2 - y_b^w\right)}{\left(g_1 - x_b^w\right)^2\left[1 + \left(\frac{g_2 - y_b^w}{g_1 - x_b^w}\right)^2\right]}, \quad \frac{\partial h_2}{\partial g_2} = \frac{1}{\left(g_1 - x_b^w\right)\left[1 + \left(\frac{g_2 - y_b^w}{g_1 - x_b^w}\right)^2\right]},$$

and $\frac{\partial h_2}{\partial g_3} = -1$.

We will not derive fully the EKF here. Please refer to Choset et. al. (2005) or more details.

4. Tests and Results

The reflective marker-based localization algorithm was implemented on the APM and tested in a variety of orchard settings. The results shown here are from tests conducted in apple orchard blocks at the Penn State University Fruit Research and Extension Center (PSU FREC) in Biglerville, PA, during June of 2009. The total area of the test block was 60 square meters, composed of six 50 meter long rows. The orchard block was a relatively new planting (~3 years old) trained in a vertical trellis system.

Reflective landmarks (traffic cones and/or reflective tape) were placed throughout the orchard block in pairs. Each pair consisted of one landmark on the left and one on the right side of the travel lane between the rows of trees. A total of 29 landmarks were placed, with pairs spaced approximately 20m apart.

The experimental procedure was to first collect a large data set that could be used to build the landmark map as described in Section 3.1. The map was then constructed offline, after which subsequent experiments were run to test our online localization algorithm, which generated real-time position estimates as the APM drove through the block. The output of the localization module was integrated with the APM graphical control panel on the laptop so that a human operator could observe localization performance during the experiment. After localization tests, results were analyzed to generate statistics that measure performance.

The result of a typical localization run is shown below in figure 9. On the left, the vehicle's path through the orchard is shown. The black line is the localization estimate from the Extended Kalman Filter, and the red line is the Applanix data, being used here for a ground truth comparison. There is also a dotted line veering off the correct path, which shows the dead reckoning estimate. The plot shows how sometimes the estimate veers off slightly from ground truth. This happens when the vehicle has travelled a significant amount of time without receiving measurements to landmarks. When a measurement is finally received, the estimate corrects itself.

The primary metric we use to analyze the error is the Euclidean distance between the estimate and the ground truth at each time step. The mean estimation error was about 20 cm with a maximum error of 1.2 m. Figure 9 (right) is a histogram showing the distribution of errors observed over the course of the experiment.

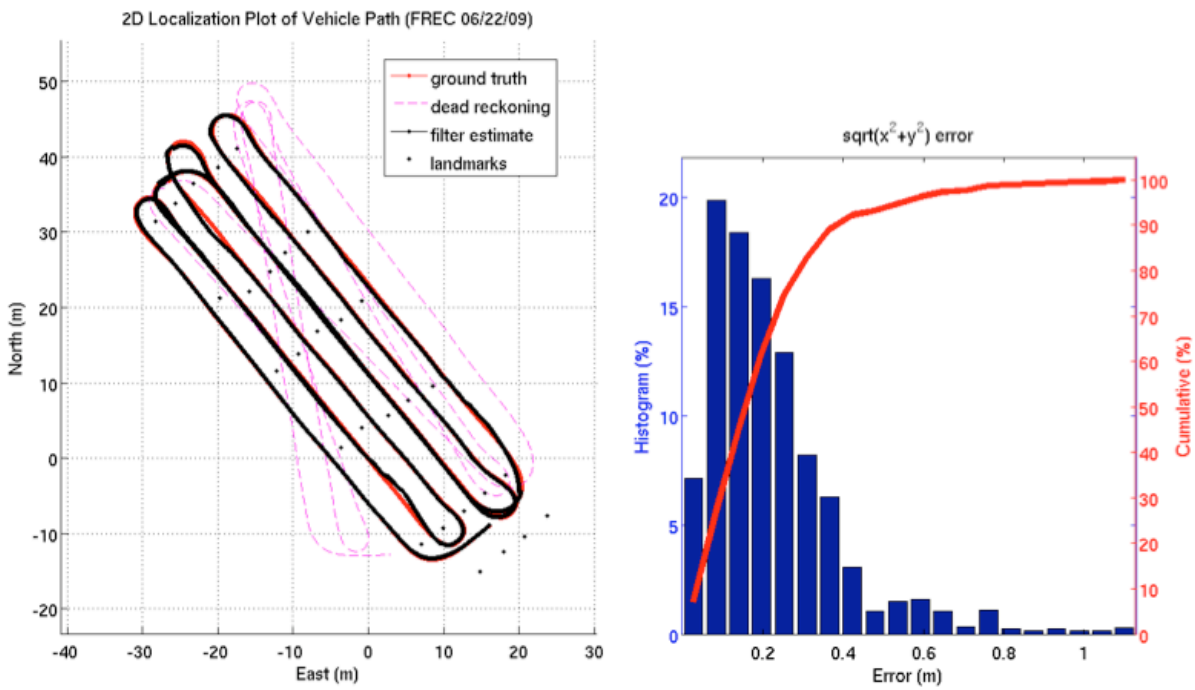

Figure 9. Localization experiment at a commercial orchard in Pennsylvania. (Left) Comparison of filter estimate, dead reckoning, and ground truth motion path. (Right) Error histogram using Euclidean distance metric.

Conclusion

We have demonstrated with our current experiments that we can attain sub-meter accuracy, using coarse odometry from the steering and wheel encoders, along with laser scanners for landmark-based localization. The primary weakness of the current implementation is that the required landmark spacing is denser than is practical for use in production nursery environments. When landmarks are placed further than 20 m apart, the dead reckoning drift becomes very large between one correction and the next. This makes it likely that the algorithm will associate the next observation to the wrong landmark, causing the filter to fail.

In ongoing work, we are tackling this problem from both ends, both by improving the prediction step of the filter (i.e. the dead reckoning) and by improving the measurement step. To improve the prediction step, we will try using laser scan matching to generate more accurate odometry measurements, reducing the rate at which the position estimate drifts when no landmarks are observed. For the measurement step, we will incorporate the use of natural features instead of the man-made reflective features that are currently being used. We will look at entire tree rows as line features, as well as individual tree trunks as point features.

Besides long stretches of travel, another point of failure for dead reckoning is when the vehicle is turning from one row to the next. Orchard rows are very narrow, challenging the turning radius limits of the vehicle. There tends to be significant wheel slip, which derails the differential drive/ Ackerman steering model discussed in section 3.2. We have already solved this problem to a large extent with landmark-based localization, and we will continue to investigate this as we strive to replace artificial landmarks with natural ones, as discussed above.

The reliance on high-accuracy GPS that we use for the mapping step is not ideal. Although the mapping is a preprocessing step that only has to happen once for a new orchard, we would like

to get rid of the need for an expensive ground-truthing system altogether. To this extent, we will investigate techniques in Simultaneous Localization and Mapping (SLAM). This will be made possible to a large extent by the incorporation of natural features, which act as more descriptive landmarks in the environment.

We will also investigate the use of other sensors, such as low-cost solid state IMU's, as well as low cost partial GPS measuring systems. The Extended Kalman Filter technique discussed here is a general sensor fusion algorithm. In this case, it is fusing the data from the encoders and the lasers. These types of filters can easily be extended to incorporate a more diverse suite of sensors. Our goal is to create low-cost robust positioning systems for outdoor environments such as specialty agriculture settings. Fusing the data from multiple cheap sensors allows our solutions to be both affordable and reliable. It also allows us to use internal and external sensors, taking advantage of the rich models we have of both our platforms and the environments in which we work.

Acknowledgements

This work is supported by the US Department of Agriculture under the Specialty Crop Research Initiative, award number 2008-51180-04876. We also thank all the members of the Carnegie Mellon University CASC team.

References

Choset, H. M., K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun. 2005. *Principles of Robot Motion: Theory, Algorithms, and Implementation*. MIT Press.

Grisettiyz, G., C. Stachniss, and W. Burgard. 2005. "Improving Grid-based SLAM with Rao-Blackwellized Particle Filters by Adaptive Proposals and Selective Resampling." *Proceedings of the 2005 IEEE International Conference on Robotics and Automation* pp: 2432-2437.

Guivant, J., H. F. Durrant-Whyte, E. Nebot. 2000. "Simultaneous Localization and Map Building Using Natural features in Outdoor Environments." In *Sixth International Conference on Intelligent Autonomous Systems*, Volume 1, pages 581–588.

Kummerle, R., B. Steder, C. Dornhege, A. Kleiner, G. Grisetti, and W. Burgard. 2009. "Large Scale Graph-based SLAM using Aerial Images as Prior Information." In *Robotics: Science and Systems*.

Murray, R. M., Z. Li, S. S. Sastry, and S. S. Sastry. 1994. *A Mathematical Introduction to Robotic Manipulation*. CRC Press.

O'Connor, M., T. Bell, G. Elkaim, and B. Parkinson. 1996. "Automatic Steering of Farm Vehicles Using GPS." In *3rd International Conference on Precision Agriculture*.

Singh, S., T. Baugher, M. Bergerman, K. Ellis, B. Grocholsky, B. Hamner, J. Harper, G. Hoheisel, L. Hull, V. Jones, G. Kantor, B. Kliethermes, H. Koselka, K. Lewis, J. Libby, W. Messner, H. Ngugi, J. Owen, J. Park, C. Seavert, W. Shi, and J. Teza. 2009. *The 4th IFAC International Workshop on Bio-Robotics, Information Technology and Intelligent Control for Bioproduction Systems.*

Spong, M. W., S. Hutchinson, and M. Vidyasagar. 2005. *Robot Modeling and Control*. Wiley.

Stentz, A., C. Dima, C. Wellington, H. Herman, and D. Stager. 2002. "A System for Semi-Autonomous Tractor Operations." *Autonomous Robots* 13, no. 1.

Thrun, S., W. Burgard, and D. Fox. 2002. *Probabilistic Robotics*. MIT Press.