Computers and Electronics in Agriculture 100 (2014) 41–50

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

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

Laser range finder model for autonomous navigation of a robot in a maize field using a particle filter

Santosh A. Hiremath [a], Gerie W.A.M. van der Heijden [a], Frits K. van Evert [b], Alfred Stein [c], Cajo J.F. ter Braak [a,*]

[a] Biometris, Wageningen UR, Wageningen, The Netherlands
[b] Plant Research International, Wageningen, The Netherlands
[c] Faculty of Geo Information Science and Earth Observation (ITC), University of Twente, Enschede, The Netherlands

ARTICLE INFO

Article history:
Received 18 January 2013
Received in revised form 12 September 2013
Accepted 12 October 2013

Keywords:
Probabilistic robotics
Autonomous navigation
Particle filter
Laser range finder

ABSTRACT

Autonomous navigation of robots in an agricultural environment is a difficult task due to the inherent uncertainty in the environment. Many existing agricultural robots use computer vision and other sensors to supplement Global Positioning System (GPS) data when navigating. Vision based methods are sensitive to ambient lighting conditions. This is a major disadvantage in an outdoor environment. The current study presents a novel probabilistic sensor model for a 2D range finder (LIDAR) from first principles. Using this sensor model, a particle filter based navigation algorithm (PF) for autonomous navigation in a maize field was developed. The algorithm was tested in various field conditions with varying plant sizes, different row patterns and at several scanning frequencies. Results showed that the Root Mean Squared Error of the robot heading and lateral deviation were equal to 2.4 degrees and 0.04 m, respectively. It was concluded that the performance of the proposed navigation method is robust in a semi-structured agricultural environment.

1. Introduction

Precision agriculture takes the variation within the field into account by observing and responding to this variation. It is considered vital for sustainable farming. Precision agriculture can be labour-intensive (Edan et al., 2009), therefore, there is great need for automation of various agricultural tasks like crop scouting, weed control, harvesting and tilling. In this vein, robotic solutions have been applied in various agricultural domains.

A basic component of automation in agriculture is autonomous navigation. Early navigation systems in agricultural domain used a camera as the sensor and were based on computer vision methods (Gerrish and Surbrook, 1984; Reid and Searcy, 1987). They were popular in agricultural robotics due to the availability of low cost cameras and the plethora of computer vision techniques that could be readily applied. For example, several methods based on the Hough transform were developed for row following (Hague and Tillett, 1996; Marchant and Brivot, 1995). Southall et al. (2002) developed a method for navigating a cabbage field in which plants were planted in a grid pattern. They used the knowledge of the environment to build a grid-based model of the local environment in the camera view to obtain the guidance information. There are also stereo based methods which try to extract depth information for robust navigation (Kise et al., 2005). Recent developments include the autonomous robots developed by Weiss et al. (2011) and Bergerman et al. (2012)

Vision based methods are sensitive to light conditions and atmospheric effects. Due to the large variation in ambient light in an outdoor environment, such as an agriculture field, most systems need frequent calibration to the specific operating conditions. Alternative methods to overcome this problems included those based on GPS technology (Heidman et al., 2002; Slaughter et al., 2008; Stoll and Kutzbach, 2001). However GPS technology has several critical drawbacks including insufficient accuracy for precision agriculture, interruptions in the signal and alterations in the environment which are not in the map but which need to be taken into account. This may lead to navigation failure.

Laser range finder(LIDAR) technology does not suffer from the effects of ambient lighting conditions and thus can be more reliable in an agricultural environment. Also the viewing range can be larger than that of a camera. Despite these advantages there is not much focus on LIDAR based navigation in agriculture mainly due to its high costs. Reducing costs in recent years has sparked renewed interest in this technology. Barawid et al. (2007) developed a real-time guidance system for navigating an autonomous vehicle in an orchard based on LIDAR. Hough Transform is used to extract plant rows for navigating the vehicle. They reported that the method is restricted to straight line recognition and thus have difficulty in curved rows. Another disadvantage of the method

* Corresponding author. Tel.: +31 317 480803.
E-mail address: cajo.terbraak@wur.nl (C.J.F. ter Braak).

0168-1699/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.compag.2013.10.005

occurs when the Hough transform fails to extract the correct plant rows causing the vehicle to lose track. LIDAR has also been used for obstacle detection and avoidance during navigation as in the case of Subramanian et al. (2006). More recently Weiss and Biber (2011) have developed a 3D LIDAR based navigation method where they used a statistical model for detection of the plant rows. The LIDAR acquires a 3D point cloud which is processed to remove the points corresponding to the ground. Then a statistical model identifies clusters of points that represents the plants. Though the results are promising, the method will not be easily scalable to other plants as the statistical model is specific to the maize plants. The statistical model depends on the cluster of 3D points which in turn depends on the shape and size of the plants. Moreover since the system is specifically designed for plant phenotyping, it imposes restrictions on the operating conditions like plant size and speed of the robot.

One of the main shortcomings in the aforementioned methods is the lack of robustness to uncertainties in the environment. Agricultural environments are dynamic and non-deterministic with several sources of uncertainty. For instance, there is noise due to uneven terrain and the varying shapes, sizes and colours of the plants. A robot operating in such an environment will suffer from wheel-slippage and sensor noise which is further compounded by controller and actuator noise. Therefore, designing a navigation method capable of managing multiple sources of variation is a challenging task. Probabilistic navigation methods proposed by Thrun et al. (2005) are most promising. They proposed a 2D LIDAR model that characterizes different types of noise in the environment. The sensor model is used within a particle filter for autonomous navigation of the robot in an indoor environment or in an outdoor urban environment.

This study aims to develop an autonomous navigation method for a robot equipped with a LIDAR for row following in a maize field. The navigation method is based on a particle filter algorithm (Thrun et al., 2005) which is used to estimate the robot-environment state of the system such as robot heading, lateral deviation, distance between the rows of plants and the end of the rows. These estimated values are in turn used to steer the robot. An important aspect of the particle filter is the measurement model. The study proposes a novel measurement model for the LIDAR where all the data obtained from the LIDAR is utilized to compute the likelihood of the particles. It is believed that this is the first (probabilistic) LIDAR model developed for robot navigation in a semi-structured environment like a maize field.

The paper is arranged as follows. Section 2 describes materials and methods along with the details of the LIDAR model. Section 3 reviews the performance and robustness of the new navigation algorithm. Whilst the limitations and extensions of the method and the scope for future research are covered in Section 4.

2. Materials and methods

2.1. Maize field

The robot navigates in a field that consists of rows of maize plants with a well defined headland. The rows may be either straight or curved. Additionally, there may be gaps within the rows. In general, the rows are approximately 0.75 m apart from each other which is the standard row width in commercial maize cultivation.

2.2. Robot architecture

The prototype robot used in this study consists of a chassis with three wheels, with overall dimensions 0.8 m × 0.45 m × 0.3 m. It has two rear wheels that do not pivot and a steering front wheel whose steering actuator is controlled via CAN-bus. All wheel units are equipped with incremental encoders to measure the rotational speed. In addition, the front wheel unit is equipped with an angle sensor to measure the steering angle. The driving speed of each wheel depends upon the target speed of the control point, the location of the wheel with respect to the control point and the turning radius. An electronic box between the rear wheels houses a mini-ITX computer with a 2.4 GHz Intel Core2 Duo processor running Windows XP operating system. The robot is controlled by a custom C# software which uses OpenCV library for image processing. Energy to the computer and the wheel units is provided by three 12 V NiMH racing packs: 1 for the front wheel unit, one for both rear wheel units, and one for the PC (see Fig. 1).

2.3. Laser range finder (LIDAR)

The robot is equipped with a LIDAR (LMS-111, Sick AG, Waldkirch, Germany) in the front at a height of 15 cm, through which it senses the world. The LIDAR operates on time-of-flight (TOF) principle. It emits pulsed laser beams using a laser diode. If a laser pulse is incident on an object, it is reflected. The reflection is detected using a photo diode. The distance to the object is calculated from the propagation time that the light requires from emission to reception of the reflection at the sensor. The emitted laser beams are deflected using a mirror at an angular resolution of 0.5 degrees and scan the surroundings in a circular manner with a maximum field of view of 270 degrees. The maximum range and scanning frequency of the LIDAR is 20 m and 50 Hz respectively.

Fig. 2(a) shows the top view of the mount. The axis of the LIDAR is aligned to the longitudinal axis of the robot. By convention, the starting and end angle of the scan are −135 and 135 degrees respectively, which are depicted by points A and C respectively in Fig. 2(a). A scan at any given time t consists of 541 observations $Z_t = (z^{(1)}, z^{(2)}, \ldots, z^{(541)})$ corresponding to the angles $\Phi = (\phi^{(1)}, \phi^{(2)}, \ldots, \phi^{(541)}) = (-135, -134.5, \ldots, 135)$, where $z^{(j)}$ is the range, that is, distance of an object (plant leaves or stem) measured by the beam j. Fig. 2(b) shows an example scan when the robot is between the rows. The data points $(\phi^{(j)}, z^{(j)})$ are represented in Cartesian coordinates for illustrative purpose. The blue circles indicate the position of the hit objects with respect to the LIDAR represented by the red circle.

2.4. Local world

A rectangular area around the centre of the robot is defined as the local world for the robot. If the robot is between the rows, the local world is approximated by two parallel rows of plants, one on either side of the robot. The rows have a finite width and are a finite distance apart. It is assumed that the row ends are usually not in view as shown in Fig. 3(a). When the robot enters into the headland, the ends of rows are in the field of view and the geometry is modelled as in Fig. 3(b). The geometry of the local world is characterized by four parameters, namely row width (rw), row distance (rd), end of left row (el) and end of right row (er). The central line halfway between the rows forms the reference axis with respect to which the robot position is determined. The robot is characterized by its main axis between the front wheel and the point halfway between the rear wheels. This point between the two rear wheels is the control point. The position of the robot in the local world is given by robot heading (h) and lateral deviation (l). The robot heading is the angle between the main axis and the reference axis measured in degrees. Lateral deviation is the signed distance between the robot's control point and the reference axis. Jointly, the parameters represent the robot-field state vector $X_t = (h, l, rw, rd, el, er)$ that characterizes the system

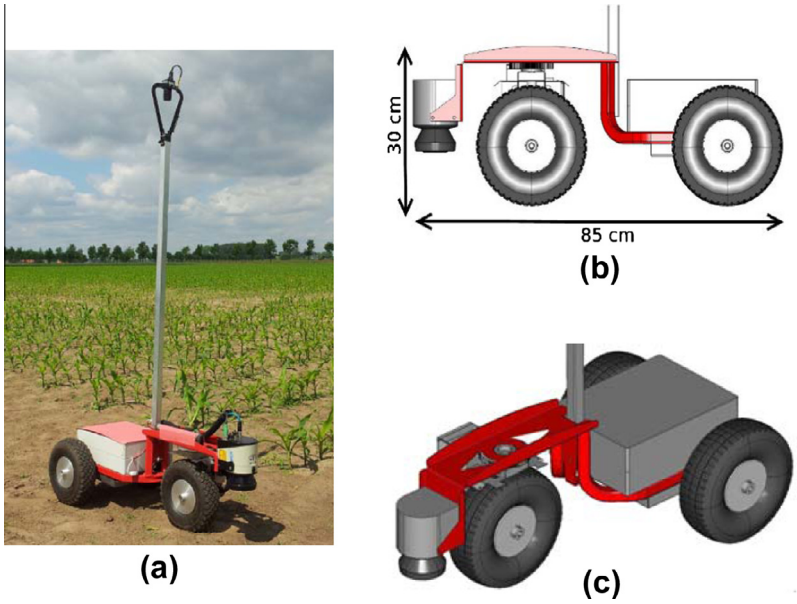

Fig. 1. (a) The robot in a field; (b) the schematic view of the robot and; (c) perspective view of the robot.

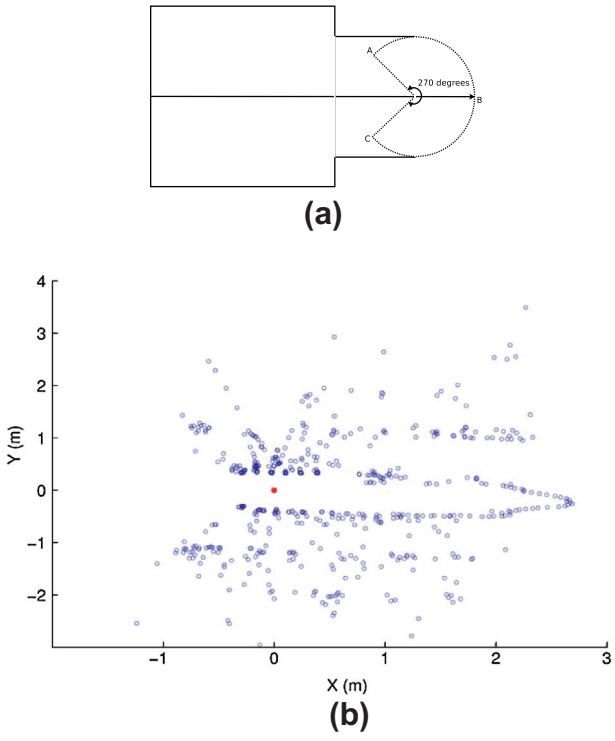

Fig. 2. (a) top view of the mounted system along with the angular convention; (b) an example scan in Cartesian coordinate system.

at a given time t. Successful navigation of the robot requires accurate estimation of the state vector at each time step.

2.5. Particle filter

The state vector denoted by $X_t = (h, l, rw, rd, el, er)$ characterizes the state of the robot in the field at any given time. Robot navigation can be considered as a dynamical system; the state changes at every time step and its values are uncertain due to different sources of noise in the environment. To deal with uncertainty the state of the system at any time is represented as a probability distribution $P(X_t|Z_{1:t}, U_{1:t})$ where $Z_{1:t}$ is the set of measurements made by the robot up to time t and $U_{1:t}$ is the set of controls applied to the robot to affect the state evolution up to time t. This distribution (also called the posterior distribution) has to be inferred at each time step. Inference of the posterior distribution is carried out by means of a particle filter algorithm. The key idea of particle filters is to represent the posterior distribution by a set of random samples called particles. These particles are recursively updated as the new measurement Z_t is acquired. The algorithm consists of two steps: prediction and update. In the prediction step the new values of the particles are calculated based on the current value and the motion model of the robot which is discussed in detail in the next section. In the update step the predicted values are evaluated for their consistency with the measurement Z_t and importance weight assigned to them. Subsequently, the particles are re-sampled according to their (normalized) importance weights to yield the posterior distribution. Formally, it is expressed as

$$p(X_{1:t}|Z_{1:t}, U_{1:t}) = \frac{p(Z_t|X_t)p(X_t|X_{t-1}, U_t)}{p(Z_t|Z_{1:t-1})} p(X_{1:t-1}|Z_{1:t-1}, U_{1:t-1}) \quad (1)$$

where, $p(Z_t|X_t)$ represents the update step and is expressed by the measurement model (also called the likelihood model or LIDAR model); $p(X_t|X_{t-1}, U_t)$ represents the prediction step and is expressed by the motion model; $p(Z_t|Z_{1:t-1})$ is the normalizing constant; and $p(X_{1:t-1}|Z_{1:t-1}, U_{1:t-1})$ is the posterior distribution at previous time step $t-1$. Details of the measurement model are discussed below. The specific form of Eq. (1) is indicative of the recursive nature of the particle filter where the posterior at the previous time step is updated by multiplying it with the motion model and the measurement model obtained from the current time step.

2.6. Motion model

The motion model describes the changes in the state vector at consecutive time steps. When the robot is between the rows, it follows the path along the centre of the plant rows. It is assumed that the distribution of the initial state (X_0 at $t = 0$) of the robot is known, and that the gyroscope and wheel encoders on the robot provide the control information $U_t = (dx, dh)$ where dx is the

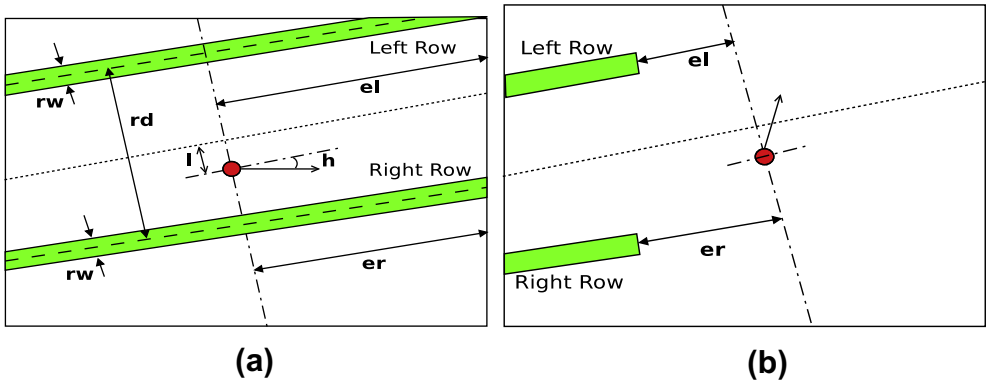

Fig. 3. The local world of the robot (a) when it is between the rows; (b) when it is within the headland. The circle represents the control point of the robot and the arrow represents the heading.

displacement of the robot along its heading and dh is the turning angle of the robot. Now, the motion model is expressed as

$$\begin{aligned} h_t &= h_{t-1} + dh + \varepsilon_h \\ l_t &= l_{t-1} + dx \ \ sin(h_{t-1} + dh) + \varepsilon_l \\ rd_t &= rd_{t-1} + \varepsilon_{rd} \\ el_t &= el_{t-1} - dx \ \ cos(h_{t-1} + dh) + \varepsilon_{el} \\ er_t &= er_{t-1} - dx \ \ cos(h_{t-1} + dh) + \varepsilon_{er} \end{aligned} \tag{2}$$

where ε_h, ε_l, ε_{rd}, ε_{el}, ε_{er} are independent Gaussian noise applied to the corresponding state variables. It is worth noting that the width of plant rows are treated as a constant. This is because the scanner data does not provide any information about the width of the rows; it only returns the distance to the first 'obstacle' that the laser ray hits.

Complication arises as the end of the rows is frequently not in view of the LIDAR. According to the motion model, values of *el* and *er* constantly decrease. When the end of row is not in robot's view, the *el* and *er* values should not be decreased. This situation is dealt with by re-initializing the *el* and *er* in a fraction of particles at regular intervals.

2.7. Measurement model

The perceptive field of the robot is modelled by a rectangular area with the LIDAR at the centre of the rectangle (the origin). When the robot is between the rows, the perceptive field is divided into five regions, named $R_1 - R_5$ based on the density of the plant material (Fig. 4). R_1 represents the soil region over which the robot travels and has the least density, R_2 and R_4 represent the region

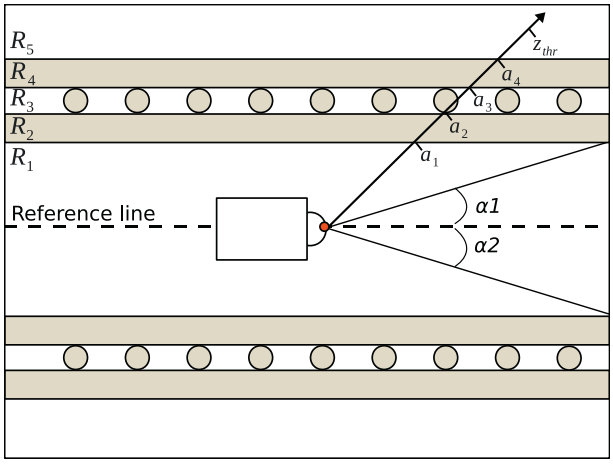

Fig. 4. Schematic diagram of the perceptive field of the LIDAR.

with foliage with intermediate density, R_3 represents the region with plant stems with highest density and R_5 represents the (soil) region beyond R_4. Regions $R_2 - R_4$ together constitute a plant row of width rw with stems along the centre line equidistant from each other at q m. The stems are assumed to be cylindrical with diameter $2r$ m.

The LIDAR makes observations by means of 541 laser beams corresponding to the 541 angles $\Phi = (-135, -134.5, -134, \ldots, 135)$. An observation is a point in the perspective field characterized by the polar coordinates (ϕ, z), where z is the range and ϕ is the angle of observation. Due to the scanning mechanism, the angle of observations is always fixed but the range varies depending on how far an object is hit (foliage or stem). Thus z is a random variable which can take value in any of the five intervals $(0,a_1]$, $(a_1, a_2]$, $(a_2, a_3]$, $(a_3, a_4]$, and (a_4, z_{thr}) corresponding to the five regions R_1, R_2, R_3, R_4 and R_5, respectively, or z falls in the category $z > z_{thr}$ where z_{thr} is a specified threshold range value. The point a_i indicates the point of crossover of the laser beam from region R_i to region R_{i+1}. The intervals are different for different observation angle ϕ and thus are different for each beam.

Due to the difference in the density of plant material in each region, the probability of a laser beam hitting plant material is also different. The region R_1 forms the robot path and objects are not expected in this region. There can, however, be unexpected objects like an occasional overhanging leaf. The likelihood of sensing such unexpected (random) objects decreases with range and thus the probability of an observation in such situations can be described mathematically by an exponential distribution (Thrun et al., 2005). Region R_2 consists of a random configuration of plant leaves. As in the case of R_1, the probability of an observation in R_2 can be described by an exponential distribution with a higher rate parameter than in region R_1 because the density of the objects in R_2 is higher than in R_1. From the assumption that plant stems in region R_3 are cylindrical of equal diameter $2r$ and at a fixed distance q from one other, the probability of a hit was derived which was converted to uniform density. The details of this derivation is included in A. Region R_4 is, like R_2, a foliage region and thus has the same exponential distribution as of R_2 and region R_5 has the same density as of R_1.

It is assumed that each observation in the measurement $Z_t = (z^{(1)}, z^{(2)}, \ldots, z^{(541)})$ is independent. Thus, the probability density of the measurement Z_t is the product of the probability densities of the individual observation and is expressed by

$$P(Z_t) = \prod_{j=1}^{541} P(z^{(j)}) \tag{3}$$

where $P(z^{(j)})$ is the probability density of the observation $z^{(j)}$. It is the probability density of a beam j hitting a plant in the perspective

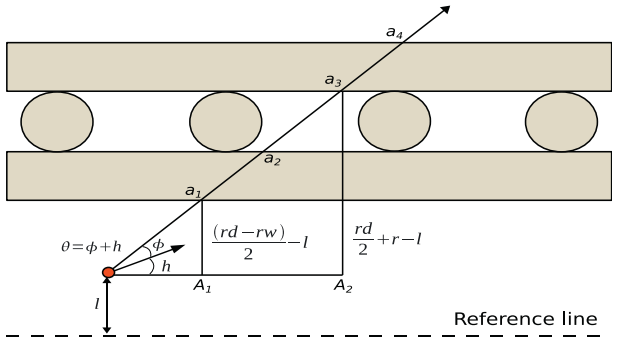

Fig. 5. Illustration of the relationship between the intervals of a beam and the state vector.

field. The probability of no-hit is the probability that the beam passes through a region without hitting anything. The probability of a hit is different in each region due to the difference in the plant density as well as due to the spatial arrangement of the regions. For instance, the probability of hit in R_2 not only depends on the plant density in that region but also on the fact that the beam does not hit any plant in region R_1 i.e., probability of no-hit of region R_1. In general, the probability of a hit in region R_i depends on the plant density of region R_i as well as the probability of no-hit of all the preceding regions through which the beam travels. Formally, it is described as follows. If an observation $x = z^{(j)}$ is in R_1, then the probability density of a hit at range x is given by

$$P(x) = \lambda_1 e^{-\lambda_1 x} \qquad (x \in (0, a_1]) \tag{4}$$

where λ_1 is the rate parameter of the exponential distribution. If $x = z^{(j)}$ is in R_2, the probability density of a hit at range x is given by

$$\begin{aligned} P(x) &= P(\text{no} - \text{hit in } R_1) \cdot P(x|\text{no} - \text{hit in } R_1) \\ &= \psi_1 \cdot \lambda_2 e^{-\lambda_2 (x - a_1)} \qquad (x \in (a_1, a_2]) \end{aligned} \tag{5}$$

where $\psi_1 = e^{-\lambda_1 a_1}$ is the probability of no-hit in R_1 and λ_2 is the rate parameter of the exponential distribution in R_2. If the observation $x = z^{(j)}$ is in R_3, the probability density of a hit at range x is

$$\begin{aligned} P(x) &= P(\text{no} - \text{hit in } R_1) \cdot P(\text{no} - \text{hit in } R_2 | \text{no} - \text{hit in } R_1) \\ &\quad \cdot P(x|\text{no} - \text{hit in} R_1, \text{no} - \text{hit in } R_2) \\ &= \psi_1 \cdot \psi_2 \cdot \left(\frac{p^*}{(a_3 - a_2)} + (1 - p^*) e^{-\lambda_2 (x - a_2)} \right) \qquad (x \in (a_2, a_3]) \end{aligned} \tag{6}$$

where $p^* = min\left(\frac{2r}{q \sin(\phi + h)}, 1\right)$, $\psi_2 = e^{-\lambda_2 (a_2 - a_1)}$ is the probability of no-hit in R_2, and q and r are the parameters of the row of stems. The derivation of p^* is given in A. Further, if the observation is in R_4, the probability density of a hit at range x is

$$P(x) = \psi_1 \cdot \psi_2 \cdot \psi_3 \cdot \lambda_2 e^{-\lambda_2 (x - a_3)} \qquad (x \in (a_3, a_4]) \tag{7}$$

where $\psi_3 = (1 - p^*) e^{-\lambda_2 (a_3 - a_2)}$ is the probability of no-hit in R_3 (A). If the beam goes beyond the four regions without hitting anything, then x is in the interval (a_4, z_{thr}). In this case the probability density of a hit at range x is given by

$$P(x) = \psi_1 \cdot \psi_2 \cdot \psi_3 \cdot \psi_4 \cdot \lambda_1 e^{-\lambda_1 (x - a_4)} \qquad (x \in (a_4, z_{thr})) \tag{8}$$

where $\psi_4 = e^{-\lambda_2 (a_3 - a_4)}$ is the probability of no-hit in R_4 and z_{thr} is a specified threshold value. Finally, the observation may be greater than or equal to the threshold value z_{thr}. For such observations, $P(z \geqslant z_{thr})$ is modelled as one minus the no-hit probability in the regions up to z_{thr}.

In order to compute the probability of hit of a beam j, it is necessary to compute its corresponding intervals $a_1, \ldots, a_4$. The variable a_i not only depends on the beam angle $\phi^{(j)}$ but also on the components of the state vector X_t as shown in the Fig. 5. This relationship enables the determination of the particle weights in the particle filter.

Fig. 6 shows the probability density profile (pdf) of a laser beam at $\phi = 45$ degrees for two sets of rate parameters λ_1 and λ_2. The assumptions for creating the plot included that the plant rows are 0.2 m wide at a distance of 0.75 m. The robot is located at the centre of the two rows with heading at 0 degree, such that the end of the rows are in sight at a distance of 0.8 m. Thus, the values of the components of the state vector are $h = 0$, $l = 0$, $rw = 0.2$, $rd = 0.75$, $el = 0.8$, and $er = 0.8$. The points a_1, a_2, a_3, a_4 and z_{thr} on the graphs indicate the boundaries of the different regions in the perspective field.

Table 1 shows the pseudocode of the measurement model although in the actual C#-code it is implemented on the logarithmic scale to prevent numerical errors. The input to the algorithm is the measurement Z_t and the particle set $\mathcal{X}_t$ and the output of the algorithm is an array W containing the likelihood of the particles in $\mathcal{X}_t$. For each observation $z^{(k)}$, first the regions $R_1, \ldots, R_5$ are computed in terms of the region bounds $a_1, \ldots, a_4$ (Line 6). Subsequently, the probability of hit is computed in Lines 8–12 based on the observation region of $z^{(k)}$. The calculation in Line 13 corresponds to the Eq. (3) which is the likelihood of the particle. Line 7 takes care of the situation when the observation is greater than z_{thr}.

2.8. Attenuation factor

The observations obtained from adjacent beams are not independent. They may be correlated as they belong to the same plant or correlation may arise due to multiple reflections of a beam from several plants. In the measurement model, however, every observation is regarded as independent. As a result the likelihood is highly peaked. That is, likelihoods of two similar particles differ by orders of magnitude. This problem is addressed by expressing the dependency between the beams by means of the attenuation factor v (Thrun et al., 2005), where v measurements are considered as a single unit reducing the effective number of measurements by a factor v. This is implemented by raising the likelihood to the power $1/v$.

The appropriate value of v is determined using the effective sample size (ESS) of the PF (Doucet and Johansen, 2009). Fig. 7 shows the plot of ESS vs v averaged over the iterations of a PF run. At small values of v, the ESS is close to 1% indicating most of the predicted particles are 'far' from the measurement and thus have negligible weights. However, when the value of v increases so does the ESS indicating that the particle weights are more evenly distributed. The value of $v = 160$ was selected so that 70% of the particles are retained after the resample step.

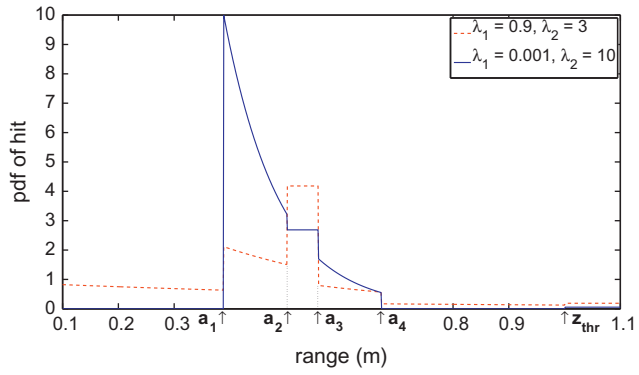

Fig. 6. Probability density of hit of a laser beam at an angle $\phi = 45$ degrees.

Table 1
Measurement model algorithm.

1	MeasurementModel($Z_t, \mathcal{X}_t$)
2	Initialize array W to hold the likelihood of the particles
3	*for each* particle x_t in the particle set $\mathcal{X}_t$ do
4	$w = 1$
5	*for* $k = 1$ to 541 *do*
6	compute region bounds $(a_1, \ldots, a_4)$ using $\phi^{(k)}$ and x_t
7	if($z^{(k)} > z_{thr}$) then $x = z_{thr}$ else $x = z^{(k)}$
8	if($x \in R_1$) then $p = \lambda_1 e^{-\lambda_1 x}$
9	if($x \in R_2$) then $p = \psi_1 \cdot \lambda_2 e^{-\lambda_2 (x-a_1)}$
10	if($x \in R_3$) then $p = \psi_1 \cdot \psi_2 \cdot \left(\frac{p^*}{(a_3-a_2)} + (1-p^*)e^{-\lambda_2 (x-a_2)}\right)$
11	if($x \in R_4$) then $p = \psi_1 \cdot \psi_2 \cdot \psi_3 \cdot \lambda_2 e^{-\lambda_2 (x-a_3)}$
12	if($x \in R_5$) then $p = \psi_1 \cdot \psi_2 \cdot \psi_3 \cdot \psi_4 \cdot \lambda_1 e^{-\lambda_1 (x-a_4)}$
13	$w = w * p$
14	*endfor*
15	$W[i] = w$
16	*endfor*
17	return W

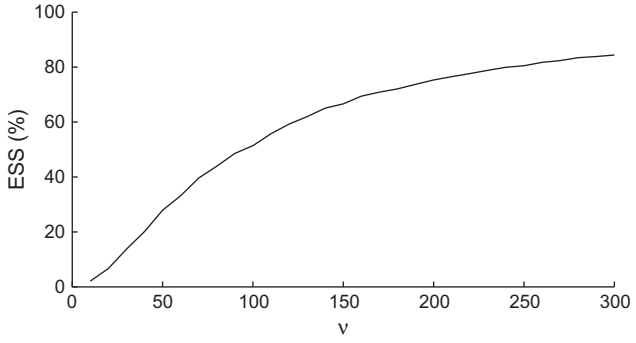

Fig. 7. The graph of *ESS* vs *v*.

2.9. Experimental data

The experimental data consists of several sequences of scans taken by the robot as it navigates through the field. The sequences are from different maize fields with different plant size and row structure (straight rows, curved rows, gaps within the rows). The number of scans in each sequence depends on the length of the robot run. The LIDAR records data at 10 Hz. Though it can record data at a maximum rate of 50 Hz, it was limited to 10 Hz for computational reasons. For every scan in the sequence, the ground truth for the most important state variables, heading (h_r), lateral deviation (l_r), end of left row (el_r) and end of right row (er_r) was established by hand annotation of a corresponding image captured by a downward looking camera mounted on the robot at a height of 1.65 m. Fig. 8 shows three example images illustrating the process of establishing the ground truth. The three images correspond to three different experimental conditions, where the plants are in different stages of development. These ground truth values were used to evaluate the performance of the PF. It should be noted that in case of large plants (Fig. 8(c)) there is some ambiguity in determining the exact position and orientation of the rows. In each picture, the yellow lines indicate the position and orientation of the plant rows, the green line indicates the robot and the red line is the reference line. The heading and lateral deviation of the robot is determined with respect to the reference line.

The test set is used for calibrating the particle filter. It consists of a single sequence of 181 scans obtained from a field with medium sized plants (Fig. 8(b)) and and field conditions in which the robot is expected to operate. The validation set, on the other hand, is used to validate the robustness of the PF. It consists of five sequences of scans obtained from different fields.

3. Experiments and results

Experiments consisted of two different phases namely, testing and validation. In the testing phase the PF algorithm and its parameters were calibrated with the test data. Subsequently, the calibrated algorithm is validated using validation data. In all the experiments, the PF was initialized with 256 particles. This number is empirically determined based on real-time computational constraints – a balance between update frequency (10 Hz) and number of particles. At each time step of the algorithm, the weighted mean of the posterior distribution X_t is the PF estimate of the state X_t at time t. Thus, for every image in the sequence, its PF estimate can be compared with the corresponding ground truth.

Calibration of the PF involved evaluating its performance on the test dataset for various combinations of rate parameters from the sets $\lambda_1 = \{0.001, 0.005, 0.001, 0.005, 0.01\}$ and $\lambda_2 = \{1, 5, 10, 15, 20, 25\}$ comparing it with the ground truth data. Fig. 9 shows the result of the comparison for three different values of λ_2 in {1, 10, 20} where $\lambda_1 = 0.005$. The five diagrams in the figure correspond to the five state variables. In this figure variables h, l, el and er are plotted along with their corresponding ground-truth data h_r, l_r, el_r and er_r, respectively, while rd had no reference. Note that $rw = 0.2$ m is assumed to be constant and hence not estimated by the PF.

The Figs. 9(a) and (b) show that the PF estimate is close to the true values of h_r and l_r when $\lambda_2 = 10$ through out the entire robot run; while in other two cases ($\lambda_2 = 1$ and $\lambda_2 = 20$) the PF estimate

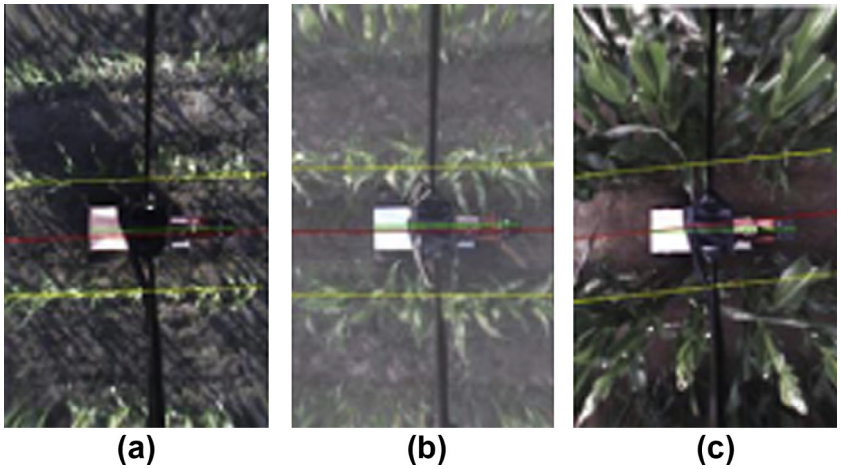

Fig. 8. Three images for three different field conditions with small plants, medium plants and large plants.

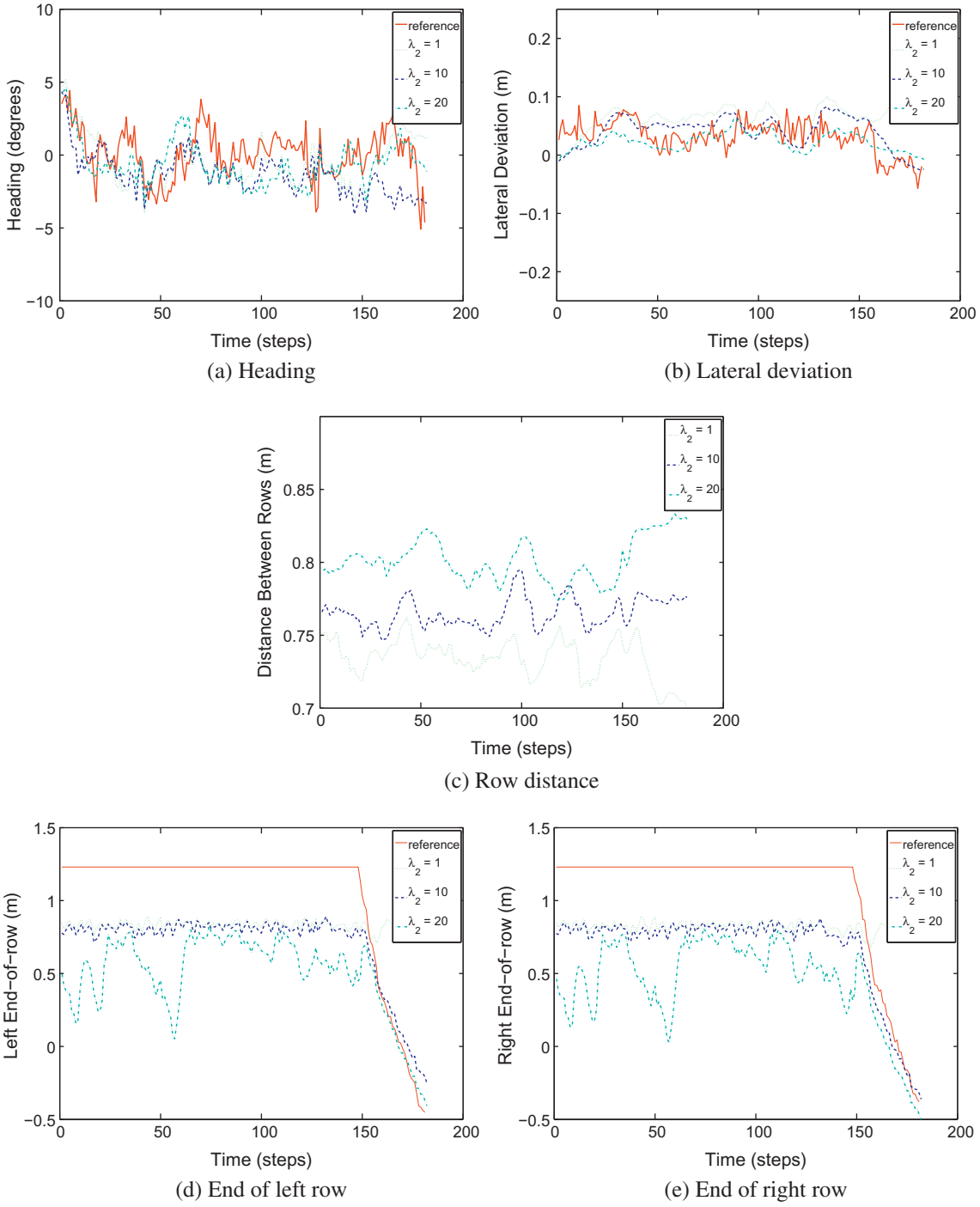

Fig. 9. The result of PF estimation of X_t of a robot run for three values of λ_2. The green dotted curve is the estimate when $\lambda_2 = 1$, the blue dashed curve is the estimate when $\lambda_2 = 10$, and the cyan dashed and dotted curve is the estimate when $\lambda_2 = 20$. The value of λ_1 was fixed to 0.001 in all three cases. The red solid line indicates the reference data. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

diverges from the ground truth at the end of the row. The detection and tracking of the end of the row is most accurate when $\lambda_2 = 10$. The end of row detection fails when $\lambda_2 = 1$ and there are many false detections when $\lambda_2 = 20$. It can also be observed (Fig. 9(c)) that the estimation of *rd* is sensitive to the values of λ_2. The increase and decrease in the value of λ_2 leads to the increase and decrease of *rd* respectively.

Based on the findings of the test phase, the calibrated PF used for the validation phase had the rate parameters $\lambda_1 = 0.005$ and $\lambda_2 = 10$. The performance of the validated PF is shown in Table 2. It gives the mean and standard deviations of the Root Mean Squared Error (RMSE) of the PF estimate of the state variables computed using the five sequences. Note that the RMSE values for *el* and *er* are determined using only the final segments of the sequences, where row ends are still in robot's view.

Other validation experiments were included to test the PF at lower scanning frequency of 5 Hz. Fig. 10 shows the comparison of the PF estimate with ground-truth data for heading and lateral

Table 2
Mean ± standard deviation of the RMSE using validation data.

Heading (degrees)	2.40 ± 1.0
Lateral deviation (m)	0.04 ± 0.02
Left end-of-row (m)	0.30 ± 0.10
Right end-of-row (m)	0.26 ± 0.10

deviation in two different field conditions. The results in the top and bottom rows correspond to the test done in the field with small (25 cm tall) and large (60 cm tall) plants, respectively.

In general, extreme values of λ_2 (and λ_1) lead to erroneous estimates of the state vector components. For instance, when $\lambda_2 = 1$ the PF fails to detect the end of the rows (green dotted curve in Figs. 9(d) and (e)). Similarly, when $\lambda_2 = 20$ there are many false detections due to small gaps from missing plants (cyan dotted and dashed curve in Figs. 9(d) and (e)). The value of λ_2 is directly proportional to the 'sensitivity' of the PF to the row ends as well as to the gaps in the row. Increasing the value of λ_2 also results in wider row distance estimates.

3.1. Model complexity

Modelling the details of the stem region R_3 increases the complexity of the measurement model. The trade-off between model complexity and algorithm accuracy was quantified by means of RMSE values of the PF estimate for *h*, *l*, *el* and *er* using the test data. The stem region R_3 can be excluded from the measurement model simply by setting $r = 0$. The experimental results are tabulated in Table 3. They suggest that the added complexity does not significantly improve the performance of the PF in a maize field. The implications of this is discussed in the next section.

4. Discussion

In this study a novel sensor model was developed for a range scanner for robot navigation in a maize field. The sensor model is used as a constituent element in a particle filter based navigation method for estimating the robot-environment state. The particle filter is able to deal with different kinds of uncertainties in the environment to provide a reliable estimate of the robot heading and lateral deviation as it moves through the field.

There is a lower limit on the plant size in which the robot can operate. This depends upon the height of the LIDAR mounted on the robot. In the experiments, the scanner was mounted at a height of 15 cm from the ground and the smallest plant size at which the algorithm gave reliable estimates was between 20–25 cm. Ideally, the plants should be 5–10 cm taller than LIDAR to satisfy the assumptions of the perspective field. Similarly, the algorithm does not perform very well when the plants are taller than 60 cm because of the dense leaves which extend across the middle of the rows and on to the main robot path. The resulting scanner data is unlike the one shown in Fig. 2(b) and the perspective field cannot easily be divided into different regions based on the differences in the density of the plants. For instance, the foliage is dense and also covers R_1. The large leaves of the plants obstruct the laser beams such that observations are clustered around the origin. Also, the measurement model cannot be used to reliably estimate the

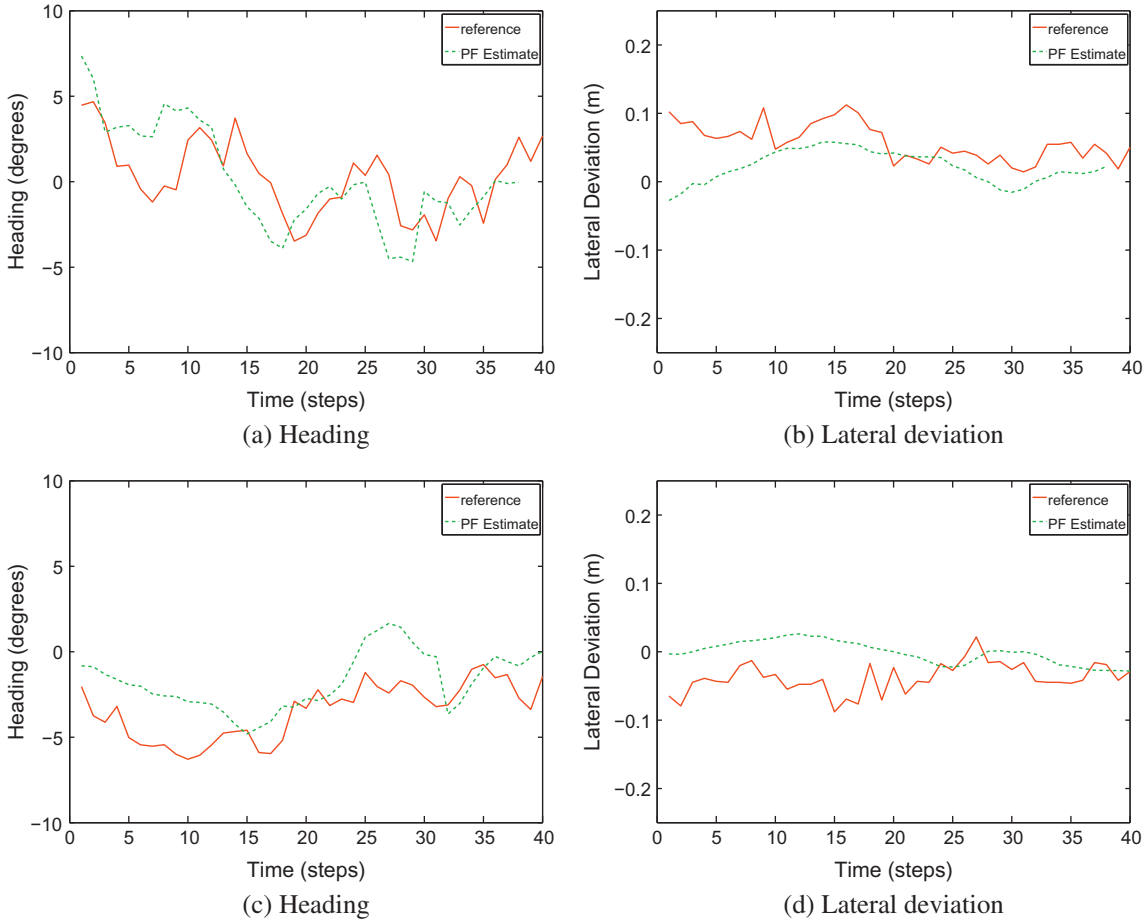

Fig. 10. The result of PF estimation of X_t of a robot run when the scanning frequency is 5 Hz.

Table 3
RMSE values using the test data.

Parameter settings	RMSE (with R_3)				RMSE (without R_3)			
	h	*l*	*el*	*er*	*h*	*l*	*el*	*er*
$\lambda_1 = 0.001$, $\lambda_2 = 1$	1.98	0.05	0.71	0.72	2.01	0.05	0.71	0.74
$\lambda_1 = 0.001$, $\lambda_2 = 5$	2	0.03	0.68	0.68	1.97	0.03	0.71	0.71
$\lambda_1 = 0.001$, $\lambda_2 = 10$	2.34	0.03	0.38	0.16	1.80	0.02	0.37	0.17
$\lambda_1 = 0.001$, $\lambda_2 = 15$	2.20	0.03	0.35	0.19	1.84	0.03	0.36	0.18
$\lambda_1 = 0.001$, $\lambda_2 = 20$	1.95	0.03	0.36	0.21	1.83	0.03	0.36	0.19
$\lambda_1 = 0.001$, $\lambda_2 = 25$	2.08	0.05	0.84	2.22	1.55	0.04	0.46	0.90

width of the plant rows. Due to the physics of the data acquisition process of the LIDAR as well as due to its positioning on the robot only the front of the object can be observed. This, however, does not affect the navigation of the robot as seen in the Section 3 where tests were carried out in fields with different plant sizes.

The process by which the ground truth is determined is based on the camera image corresponding to a scan. The difference in the camera's downward-looking and the LIDAR's forward-looking perspectives results in larger RMSE values for the end of row components (*er* and *el*) than that was observed in field experiments. This is because, in the camera image, the end of a row is determined by the tip of leaf extending furthest into the headland which may not be accurate from a LIDAR perspective. The heading and lateral deviation components does not suffer from this perspective bias. The problem is further compounded by hardware errors which result in missing scans.

The experiments showed that the added complexity of the stem region (R_3) does not improve the accuracy of PF estimation. This may be explained by the fact that the density of the foliage is greater than the density of the stems for maize plants. A closer examination of Fig. 6 shows that, in the experimental situation, the foliage density is greater than the density of the stem region (the solid blue curve where $\lambda_2 = 10$). Here the exponential density of the R_2 dominates over the density of the stem region R_3. This effect may not hold for other crops as indicated by the dashed red curve which corresponds to a situation where the density of the stems in R_3 is greater than the foliage density in R_2. In conclusion, the inclusion/exclusion of a particular region depends on the crop, in which the robot will operate and the parameters of the measurement model can be tuned as necessary.

The scanning rate impacts the performance of the algorithm. Low scan rate leads to erroneous PF estimate, while a high scan rate improves the accuracy of the PF estimate. This, however, increases the computational burden and thus the trade off between the two is a design decision based on the application. As demonstrated in Section 3, scanner data acquired at a rate of 10 Hz is sufficient for robust estimation of the robot-environment state.

The proposed LIDAR model is closely related to the beam model developed by (Thrun et al., 2005). The beam model is a mixture of four densities where each density characterizes a type of noise typically encountered when using a LIDAR. In comparison, the different types of noise in the beam model is analogous to the different regions in the measurement model. The two models differ in the way in which the hit probability of the beam is computed. In the measurement model, it depends on the spatial arrangement of the regions as well as the sequence in which the beam encounters these regions; whereas the beam model is simply a weighted average. A main advantage of the measurement model is that it takes in to consideration the geometry of the perspective region and the angles of the laser beams unlike the beam model.

Robot navigation in a maize field using a laser range finder is not a trivial task because range data are noisy. The noise may be due to incorrect observations due to multiple reflections, missing observations or sometimes even missing scans. A probabilistic sensor model is a good way to characterize these various types of sensor noise. As far as it is known this is the first probabilistic 2D LIDAR model for robot navigation in a maize field. The sensor model can be easily extended to included more regions within the perspective field if necessary. Each region can be of a new type with its own probability distribution. As a result the model can be applied in other fields.

5. Conclusion

This study focused on developing a probabilistic navigation method based on a particle filter (PF). Such a probabilistic approach mitigates the effects of uncertainties in the environment. This is demonstrated in the performance of the PF under various field conditions, where it provided accurate estimates of the robot-environment state in a dynamic and noisy environment. The probabilistic model of the LIDAR sensor incorporated in the PF turned out to be an effective way of dealing with sensor noise. Because of its probabilistic nature, the model could easily be incorporated into the PF. Using a LIDAR sensor for sensing the world also expanded the operating conditions of a robotic system in an agricultural field as, for example, it did not suffer from the effects of varying lighting conditions. In conclusion, it was shown that using LIDAR with a Particle Filter enabled robust robot navigation within the semi-structured agricultural environment of a maize field.

Acknowledgements

The authors thank Mr. Martijn van der Bijl and colleagues of Kverneland Group, Nieuw Vennep, The Netherlands, for providing the robot, including the microcontroller (low-level control) software. The authors are indebted to Mr. Arjan Lamaker, Wageningen, The Netherlands, for writing the PC (high-level control) software. Sick B.V., Bilthoven, The Netherlands, supported this work by providing a LMS-111 LIDAR.

Appendix A. Probability of an observation in R_3

The top view of a section of the stem region R_3 is shown in Fig. A.11. It consists of a series of cylindrical stems of diameter $2r$ with centres at a distance of q from one stem to the next. Consider a laser beam j at an angle ϕ. S1 and S2 represent two adjacent

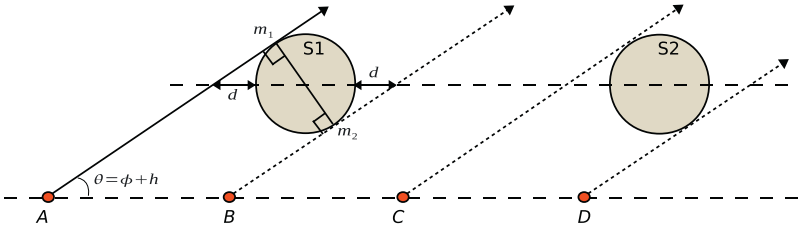

Fig. A.11. A section of the stem region.

stems of diameter $2r$ and with centres at a distance q apart. The red circles indicate the position of the LIDAR. A laser beam j with angle ϕ strikes S1 between the points m_1 and m_2. The line through A and m_1 is tangent to S1 and so is the line through B and m_2. The beam will strike the stem S1 between the points m_1 and m_2 when the LIDAR is between the positions A and B, respectively, where m_1 is the first pointand m_2 is the last point of contact. The same beam will strike stem S2 when the LIDAR is between C and D. Assuming that the stems are uniformly random with respect to the position of the robot, the probability of a beam striking a stem is related to the area covered by the stems. In other words it is proportional to the ratio $\frac{2r+2d}{q}$. From trigonometry $d = r\left(\frac{1}{sin(\theta)} - 1\right)$ where $\theta = \phi + h$ and thus the ratio can be rewritten as $\frac{2r}{q\,sin(\theta)}$. The probability of a stem hit is thus

$$p^* = min\left(\frac{2r}{q\,sin(\theta)}, 1\right)$$

If the beam does not hit a stem (probability $1 - p^*$), it may still hit foliage as in R_2. Assuming a uniform density for a stem hit, the final pdf in region R_3 is

$$P(x) = \frac{p^*}{a_3 - a_2} + (1 - p^*)e^{-\lambda_2(x - a_2)}$$

Integrating this over the range $(a_2, a_3]$ gives the probability of hit with the resulting probability of no-hit in R_3 being $\psi_3 = (1 - p^*)e^{-\lambda_2(a_3 - a_2)}$.

References

Barawid Jr., O.C., Mizushima, A., Ishii, K., Noguchi, N., 2007. Development of an autonomous navigation system using a two-dimensional laser scanner in an orchard application. Biosystems Engineering 96 (2), 139–149.

Bergerman, M., Singh, S., Hamner, B., 2012. Results with autonomous vehicles operating in specialty crops. In: IEEE International Conference on Robotics and Automation (ICRA), 2012, pp. 1829–1835.

Doucet, A., Johansen, A.M., 2009. The Oxford Handbook of Nonlinear Filtering. Oxford University Press, Chapter: A Tutorial on Particle Filtering and Smoothing: Fifteen Years Later.

Edan, Y., Han, S., Kondo, N., 2009. Automation in Agriculture, in Handbook of Automation. Springer.

Gerrish, J., Surbrook, T., 1984. Mobile robots in agriculture. In: First International Conference on Robotics and Intelligent Machines in Agriculture,1984, St. Joseph, MI, American Society of Agricultural Engineer, pp. 30–41.

Hague, T., Tillett, N., 1996. Navigation and control of an autonomous horticultural robot. Mechatronics 6 (2), 165–180.

Heidman, B., Abidine, A., Upadhyaya, S., Hills, D., Robert, P., 2002. Application of RTK GPS based auto-guidance system in agricultural production. In: Proceedings of the 6th International Conference on Precision Agriculture and Other Precision Resources Management, 2002, Minneapolis, USA, pp. 1205–1214.

Kise, M., Zhang, Q., Mas, F., 2005. A stereovision-based crop row detection method for tractor-automated guidance. Biosystems Engineering 90 (4), 357–367.

Marchant, J., Brivot, R., 1995. Real-time tracking of plant rows using a hough transform. Real-Time Imaging 1 (5), 363–371.

Reid, J., Searcy, S., 1987. Vision-based guidance of an agricultural tractor. IEEE Control Systems Magazine 7 (2), 39–43.

Slaughter, D., Giles, D., Downey, D., 2008. Autonomous robotic weed control systems: a review. Computers and Electronics In Agriculture 61 (1), 63–78.

Southall, B., Hague, T., Marchant, J.A., Buxton, B., 2002. An autonomous crop treatment robot: Part i. A Kalman filter model for localization and crop/weed classification. International Journal of Robotics Research 21 (1), 61–74.

Stoll, A., Kutzbach, H., 2001. Guidance of a forage harvester with gps. Precision Agriculture 2 (3), 281–291.

Subramanian, V., Burks, T., Arroyo, A., 2006. Development of machine vision and laser radar based autonomous vehicle guidance systems for citrus grove navigation. Computers and Electronics in Agriculture 53 (2), 130–143.

Thrun, S., Burgard, W., Fox, D., 2005. Probabilistic Robotics. The Mit Press.

van Evert, F.K., van der Bijl, M., Lamaker, A., Stravers, T., Polder, G., van der Heijden, G.W., Kroon, B., Knol, J., Dhaene, M., van der Zalm, A., Bakker, T., Lotz, L., 2011. Hugo. In: Proceedings of the 8th Field Robot Event, 2010, Braunschweig, Germany, pp. 88–99. <http://www.digibib.tu-bs.de/?docid=00041345>.

Weiss, U., Biber, P., 2011. Plant detection and mapping for agricultural robots using a 3d lidar sensor. Robotics and Autonomous Systems 59 (5), 265–273.