Image-Based Particle Filtering For Robot Navigation In A Maize Field

Santosh Hiremath[1], Frits van Evert[2], Gerie van der Heijden[3], Cajo ter Braak[3] and Alfred Stein[4]

***Abstract*— Autonomous navigation of a robot in an agricultural field is a challenge as the robot is in an environment with many sources of noise. This includes noise due to uneven terrain, varying shapes, sizes and colors of the plants, imprecise sensor measurements and effects due to wheel-slippage. The drawback of current navigation systems in use in agriculture is the lack of robustness against such noise. In this study we present a robust vision-based navigation method based on probabilistic methods. The focus is on navigation through a corn field. Here the robot has to navigate along the rows of the crops, detect the end of the rows, navigate in the headland and return in another row. A Particle Filter based navigation method is used based on a novel measurement model. This model results in an image from the particle state vector that allows the user to compare the observed image with the actual field conditions. In this way the noise is incorporated into the posterior distribution of the particle filter. The study shows that the new method accurately estimates the robot-environment state by means of a field experiment in which the robot navigates through the field using the particle filter.**

I. INTRODUCTION

In recent years robots are being used for automating several agricultural operations including harvesting [1], scouting [2] and weed control [3], [4]. A desired quality of such robots is the ability to navigate autonomously without manual intervention. Most of the existing systems are vision based which include [5]–[7].

As most crops are cultivated in rows, research in autonomous navigation in agriculture has focused on navigation systems that operate within rows of plants. A crucial component of such a navigation system is the ability to detect the rows. This problem is addressed by extracting line features from the image representing the plant rows which are used as navigation cues to steer the robot. A commonly used method for extracting lines is the Hough transform [8]–[10]. The Hough transform, however, is not robust to uncertainty in the environment and fails to extract the 'correct' lines, leading to navigation failure. Although, additional heuristics may improve the results [11], [12] it is insufficient to account for the uncertainty in the environment. The main problem of the Hough transform as well as of other line extraction methods is its failure in line extraction procedures that leads to problems in navigation from which the algorithm cannot recover. Moreover, adaptation to curved or irregular plant rows is not straightforward.

In this research we address autonomous navigation of a field robot from a probabilistic perspective. It is an extension of the work in [13]. Due to irregularity in field conditions, any line extraction procedure is likely to fail at some point. For that reason it will be beneficial if the navigation algorithm has the property to recover from such failures. We adopt a framework proposed by [14] where we use a particle filter to track multiple hypotheses about the position of the rows and the location of the robot relative to them. In this framework, the state of the robot and the field at time t is represented by a probability distribution $P(X_t|Z_{1:t},U_{1:t})$ where X_t characterizes the state of the robot and the field with which it interacts via its sensor, $Z_{1:t}$ represents the sensor measurements up to time t and $U_{1:t}$ the controls applied to the robot up to time t. The particle filter algorithm estimates $P(X_t|Z_{1:t},U_{1:t})$ by maintaining a set of samples from the distribution called particles. The set is updated at each time step based on the current measurement Z_t. Commonly, Z_t represents extracted line features from the camera image. The feature extraction process itself, however, introduces uncertainty that cannot be handled within the particle filter framework. To address this problem this paper introduces model images from the particles that are used as predictions. We compare these predictions with the actual measurement Z_t to update the probability distribution.

The paper is organized as follows. Section II describes the essential components of the robot hardware relevant to the presented work. Sections III and IV describe the field in which the robot operates and the local world of the robot (based on the camera view) within it, respectively. Section V details the overall navigation of the robot. In section VI, we give the details of the image-based particle filter algorithm. Section VII describes the image processing steps used to obtain Z_t and finally section VIII shows some results.

II. ROBOT HARDWARE

The robot that is employed in this study consists of a chassis with three wheels. It has an actuated front wheel as the steering wheel that is affected by commands from a control program by means of a CAN-bus and it has two rear wheels that do not have the ability to steer. 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.

[1]Biometris, Wageningen University and Research Centre, Droevendaalsesteeg 1, 6708 PB Wageningen, Netherlands santosh.hiremath@wur.nl

[2]Plant Research International, Droevendaalsesteeg 1, 6708 PB Wageningen, Netherlands

[3]Biometris, Wageningen University and Research Centre, Droevendaalsesteeg 1, 6708 PB Wageningen, Netherlands

[4]Faculty of Geo Information Science and Earth Observation (ITC), University of Twente, Netherlands

The robot senses the field by means of a downward-looking camera (uEye UI-1220 SE, IDS Imaging Development Systems GmbH, Obersulm, Germany) with a 2.4 mm, 186 degrees field-of-view lens CF2420 (Lensation GmbH, Karlsruhe, Germany) that is mounted at a height of 1.65 m. A gyroscope (Inertia-Link, Microstrain Inc., Williston VT, USA) provides information about the rotational speed of the robot. The robot further has a laser scanner (LMS-111, Sick AG, Waldkirch, Germany) in the front but it is not used in the current study (Figure 1).

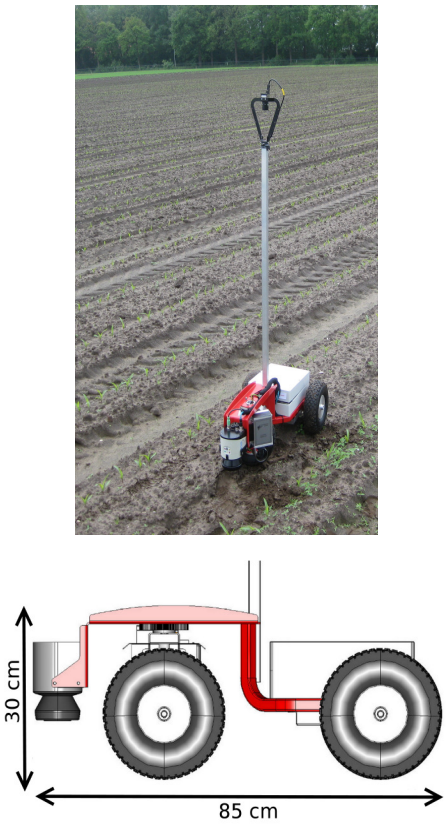

Fig. 1: The top figure shows the robot with the mounted camera in a field. The bottom figure shows the profile view of the robot drawn to scale. Clearly visible in the design is the actuated front wheel that can be steered.

III. MAIZE FIELD

The robot navigates in a field that consists of rows of maize plants with a well defined headland. These rows may be either straight, curved or jagged. This also applies to the path along the headland, which can be either perpendicular to the plant rows or at an angle to them. Additionally, there may be gaps within the rows. In general, the rows are approximately 0.75 m apart, being the distance at which the seeding took place.

IV. LOCAL WORLD

To ensure that robot navigation is not affected by different types of row patterns in the field, a rectangular area around the robot is defined as the local world with the robot at the centre. Based on the camera view, if the robot is between the rows, the local world is approximated by two parallel rows of plants on either side of the robot with finite width and at a finite distance apart. Usually, the row ends are out of the camera view. Figure 2a shows the local world in this situtaion. Similarly, when the robot enters the headland, the end of rows are in the field of view and the geometry is modelled as in Figure 2b. The geometry can be characterized by row width (rw), row distance (rd), end of left row (el) and end of right row (er), all measured in meters. The robot is characterized by its main axis along the direction of travel and a central control point half way between the wheels. 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, being the line along the centre of the rows, measured in degrees. Lateral deviation is the 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 at a given time t. Successful navigation of the robot requires accurate estimation of the state vector at each time step.

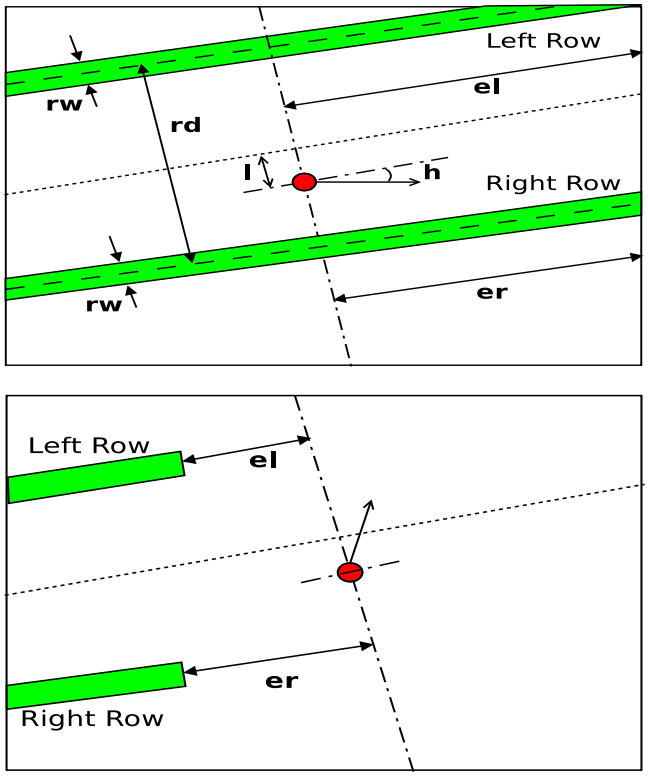

Fig. 2: The local world of the robot when it is between the rows (top) and when it is within the headland (bottom). The red circle with the arrow represents the robot, with the circle representing the control point of the robot and the arrow its heading.

V. NAVIGATION

The task of the robot is to navigate along plant rows, detect the end of the rows, navigate on the headland and return into another row. Navigation of the robot is affected by the robot controller which uses the state vector estimates from the particle filter (see Section VI). This can be divided into two distinct situations: navigation between the rows and on the headland. As long as the robot is between the rows, it follows the in-row reference line. As soon as it is on the headland, however, it follows the headland reference line as shown in Figure 3. In either case the target steering angle of the front wheel unit is given by $\gamma = -h + tan^{-1}(-a/b)$ where h is the robot heading, a is the perpendicular distance of the robot from the reference line and b is the target distance along the reference line. Note that $a = l$ if the robot is between the rows.

After detecting the end of the row, the robot continues following the rows until its control point is at a given distance from the row end. It then makes an on-the-spot turn to position itself parallel to headland reference line and continues the headland navigation. Upon reaching the middle of the next row, it comes to a full stop, makes an on-the-spot turn to position itself parallel to the rows and starts following the rows again. Before the robot begins travelling in the new row, the coordinate system is reset such that the centre of the current row is the in-row reference line.

An important element of navigation on the headland is counting how many rows have been crossed. To do so, a row counter is used that is initially set to zero and is updated when the lateral deviation (l) of the robot is larger than half the row distance (rd).

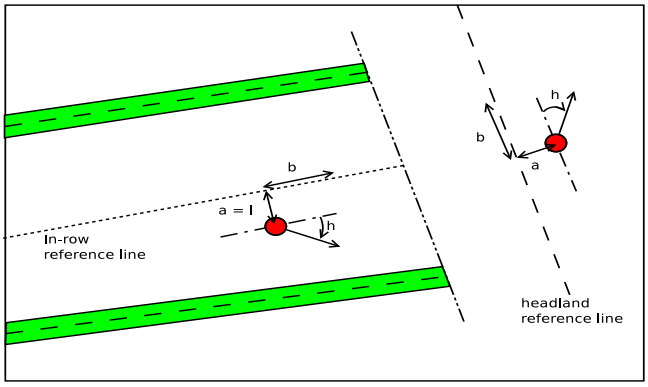

Fig. 3: The target steering angle γ as it is computed by the controller. Between rows the robot follows the in-row reference line (dotted line) and within the headland the robot follows the headland reference line (dashed line).

VI. PARTICLE FILTER

The robot-field 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. The robot can be considered as a dynamical system in which the state changes at every time step and where the values of the state vector are uncertain due to inherent irregularities in the field conditions. Several sources of uncertainty can be distinguished, in particular uneven terrain conditions, inconsistent field structure and varying shape, size and colour of the plants. Additional uncertainty is due to errors in sensor measurements and hardware related errors like wheel-slippage or controller and actuator noise. To deal with all this uncertainty we represent the state of a system at any given time 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. It is also called the posterior distribution. Stated this way, the posterior distribution has to be inferred at each time step t. This inference problem is shown graphically in Figure 4.

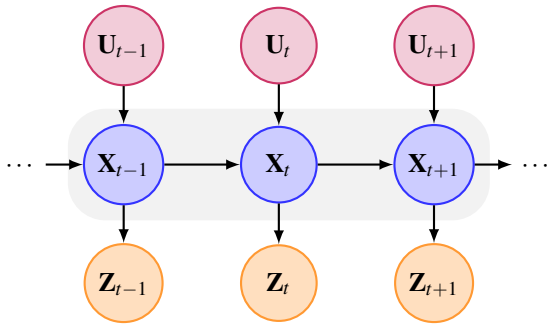

Fig. 4: Graphical representation of the inference problem for $P(X_t|Z_{1:t}, U_{1:t})$. The nodes labeled X_t, Z_t and U_t represent the state, measurement and controls at time t, respectively.

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 drawn from this distribution, called particles. These particles are recursively updated as a 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 that we will discuss 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 given by

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

where, $p(Z_t|X_t)$ represents the update step and is given by the measurement model (also called the likelihood model), $p(X_t|X_{t-1}, U_t)$ represents the prediction step and is given by the motion model, $p(Z_t|Z_{1:t-1})$ is the normalizing constant, and $p(X_{t-1}|Z_{1:t-1}, U_{1:t-1})$ is the posterior distribution at previous time step $t-1$. Details of the measurement model will also be discussed below. The specific form of equation

1 is indicative of the recursive nature of the particle filter where the posterior at previous time step is updated by multiplying it with the motion model and the measurement model obtained from the current time step.

A. MOTION MODEL

The motion model describes how the state vector changes from one time step to another. Here we assume 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 displacement of the robot along its heading and dh is the turning angle of the robot. The motion model is then given by

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

where $\varepsilon_h, \varepsilon_l, \varepsilon_{rw}, \varepsilon_{rd}, \varepsilon_{el}, \varepsilon_{er}$ are independent Gaussian noise applied to the corresponding state variables.

A complication arises because the end of the rows is frequently not in view of the robot's camera. According to the motion model, the values of el and er are constantly decreased. 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 el and er in a fraction of particles at regular intervals.

B. MEASUREMENT MODEL

The measurement Z_t is a binary image of size $m = r \times c$ pixels which is obtained after processing the camera image. The details of the image processing steps are given in section VII. Pixels with value one ($z_i = 1$) indicate plants; pixels with value zero ($z_i = 0$) indicate soil. The measurement model for Z_t depends on a model image. The model image is a binary image constructed from a particle. The model image consists of two regions: the in-row region and the out-row region. The in-row region is the predicted location of the plant rows and is a region with a high probability of having plants present. The out-row region is the region with a lower probability. The likelihood function assigns an importance weight to the model image-based on how well it 'agrees' with the observed image. The importance weights are used in the re-sampling step.

VII. IMAGE PROCESSING

The steps during image processing to obtain the measurement image Z_t are illustrated in Figure 5 that shows the transformation of the input image at various processing stages. The input image (Figure 3a) captured by the camera is of size $width \times height = 752 \times 480$ pixels where height is in the travelling direction and width is perpendicular to it. The

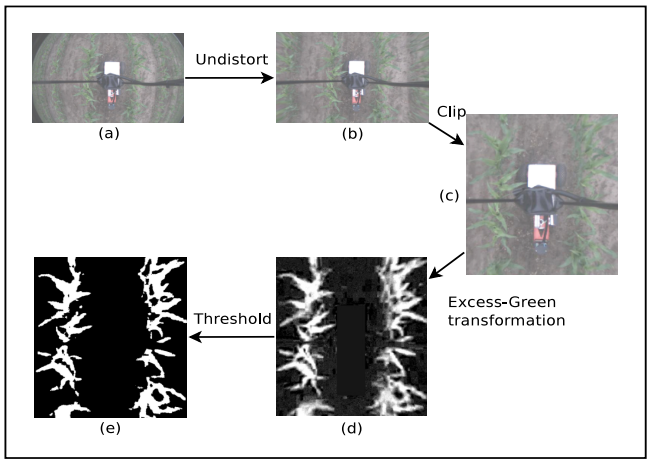

Fig. 5: Transformation of the input image during different image processing steps, from the input image (a), through the undistorted image (b), the clipped image (c) and the excess-green image (d) towards the final binary image after thresholding (e)

first processing step corrects the barrel distortion in the input image resulting in the undistorted image (Figure 3b). This image is then clipped to a size of 40×70 pixels such that only the two rows adjacent to the robot remain. The clipped image (Figure 3c) is subsequently converted to an excess-green image (Figure 3d) by applying the transformation $I = (-R + 2G - B)/3$. Finally, by thresholding the measurement image is obtained (Figure 3e). The image processing method depends on the plants being green. Specifically, the excess-green transformation exploits the colour contrast between maize plants and the soil. It can be extended to plants not as green as maize by adjusting the coefficients of the colour channels.

When the robot is between the rows the clipped image is 'long' in the sense that the image width is less than the image height. This ensures that only the essential information captured by the camera is retained to accurately determine the orientation of the rows. In the headland, however, a 'wide' image ensures that enough of the row is visible to determine its orientation accurately (see Figure 6). Hence, the second image processing step is different when the robot is in the headland to get a clipped image of size 105×60 pixels.

VIII. RESULTS

In this section we present some results. At each time step, the mean of the posterior distribution $\hat{X}_t$ is used as the estimate of the state X_t at time t. Figure 7 shows examples of the estimate superimposed on the observation Z_t. Pixels in red indicate the estimated plant rows and the pixels in green are the observed rows. As we see in Figure 7a, the estimate is consistent with the observation when the robot is in between the rows. Likewise, Figure 7b shows the accurate detection of the row ends when the robot is in the headland.

Fig. 6: Clipped image in the headland. The image is 'wide' relative to clipped image when the robot is between the rows

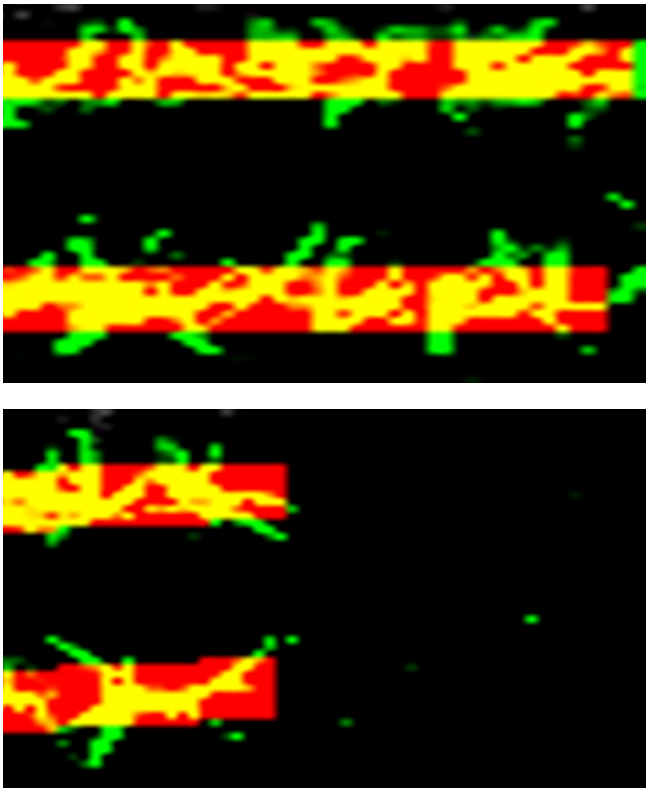

Fig. 7: Model image of the mean particle superimposed on the observed image when the robot is between the rows (top) and when the robot is in the headland (bottom).

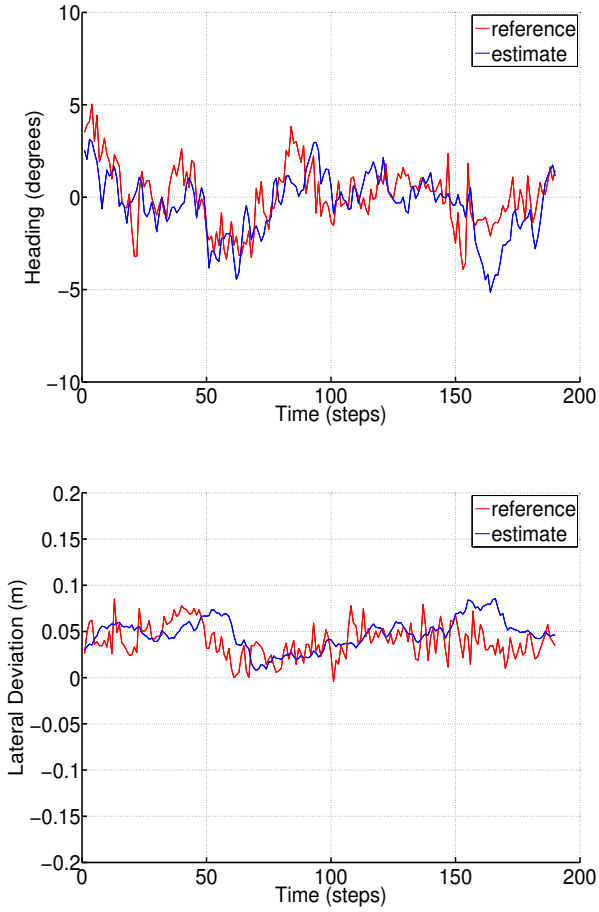

Fig. 8: The result of particle filter estimate of heading (top) and lateral deviation (bottom) for one robot run along with the manually estimated values. The red curve is the ground truth and the blue curve is the particle filter estimate.

Figure 8 shows results of a single run by the robot within a maize field. An image sequence is collected that consisted of all images captured at a rate of 10 Hz starting from the beginning of a single row to the end of that row where an image in the sequence corresponds to a time step t in the particle filter. For every image in the sequence, the estimate of the system state is compared with the ground truth that is obtained manually. Figure 8a shows the plot of robot heading (in degrees) against time t where the blue curve represents the estimated values and the red curve represents the ground truth values. As we can see, the particle filter accurately estimates the robot heading with respect to the plant rows. Similarly, Figure 8b shows the plot of lateral deviation (in meters) against time t. The figures indicate that there is greater variation in robot heading than in lateral deviation. This is because of the system design. That is, change in lateral deviation can only be affected through a combination of rotation and translation. Also, the front wheel constantly receives steering signals from the controller and thus have greater susceptibility to controller and actuator noise.

IX. DISCUSSION

Field experiments showed that the robot was able to navigate without manual intervention until the batteries were empty. Experiments were carried out in fields from young plants of 5cm to plants of over 50cm, in straight rows and curved rows. These experiments showed that the new image-based particle filter is robust to different types of uncertainties in the field.

When the plants were too big or when the soil between the rows is covered with weeds, the measurement image consists of only plants. In this situation the algorithm fails because it

is difficult to extract the rows. We intend to address this problem in future research by integrating the laser range sensor data to the particle filter algorithm. By combining both camera and laser range data we hope to widen the operating conditions of the navigation system.

REFERENCES

[1] N. Kondo and K. Ting, "Robotics for bioproduction systems," *Robotics For Bioproduction Systems*, 1998.

[2] S. Pedersen, S. Fountas, H. Have, and B. Blackmore, "Agricultural robots - system analysis and economic feasibility," *Precision Agriculture*, vol. 7, no. 4, pp. 295–308, 2006.

[3] D. Slaughter, D. Giles, and D. Downey, "Autonomous robotic weed control systems: A review," *Computers And Electronics In Agriculture*, vol. 61, no. 1, pp. 63–78, 2008.

[4] F. K. van Evert, J. Samsom, G. Polder, M. Vijn, H.-J. van Dooren, A. Lamaker, G. W. A. M. van der Heijden, C. Kempenaar, T. van der Zalm, and L. A. P. Lotz, "A robot to detect and control broad-leaved dock (rumex obtusifolius l.) in grassland," *Journal of Field Robotics*, vol. 28, no. 2, pp. 264–277, 2011.

[5] J. N. Wilson, "Guidance of agricultural vehicles - a historical perspective," *Computers and Electronics in Agriculture*, vol. 25, no. 1-2, pp. 3–9, 2000.

[6] R. Keicher and H. Seufert, "Automatic guidance for agricultural vehicles in Europe," *Computers and Electronics in Agriculture*, vol. 25, no. 1-2, pp. 169–194, 2000.

[7] J. F. Reid, Q. Zhang, N. Noguchi, and M. Dickson, "Agricultural automatic guidance research in North America," *Computers and Electronics in Agriculture*, vol. 25, no. 1-2, pp. 155–167, 2000.

[8] J. A. Marchant, "Tracking of row structure in three crops using image analysis," *Computers And Electronics In Agriculture*, vol. 15, no. 2, pp. 161–179, 1996.

[9] F. Rovira-Ms, Q. Zhang, J. F. Reid, and J. D. Will, "Hough-transform-based vision algorithm for crop row detection of an automated agricultural vehicle," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 219, no. 8, pp. 999–1010, 2005.

[10] G. Jiang, C. Zhao, and Y. Si, "A machine vision based crop rows detection for agricultural robots," in *2010 International Conference on Wavelet Analysis and Pattern Recognition*, 2010, pp. 114–118.

[11] V. Leemans and M. Destain, "Line cluster detection using a variant of the hough transform for culture row localisation," *Image and Vision Computing*, vol. 24, no. 5, pp. 541–550, 2006.

[12] G. Jiang, X. Ke, S. Du, and J. Chen, "Detection algorithm of crop rows based on machine vision and randomized method," *Nongye Jixie Xuebao/Transactions of the Chinese Society of Agricultural Machinery*, vol. 39, no. 11, pp. 85–88+93, 2008.

[13] F. v. Evert, A. Bijl, M. van derAnd Lamaker, T. Stravers, G. Polder, G. v. d. Heijden, B. Kroon, J. Knol, M. Dhaene, A. v. d. Zalm, T. Bakker, and L. Lotz, "Hugo," in *Proceedings of the 8th Field Robot Event 2010, Braunschweig, Germany, 11-13 July 2010, Braunschweig: Technische Universitat Braunschweig*, 2011, pp. 88–99.

[14] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics And Autonomous Agents)*. The Mit Press, 2005.