

# Particle Filter Based Outdoor Robot Localization Using Natural Features Extracted from Laser Scanners

Martin Adams, Sen Zhang and Lihua Xie  
*School of Electrical and Electronic Engineering*  
*Nanyang Technological University, Singapore 639798*  
{eadams, ps7184247i, elhxie}@ntu.edu.sg

**Abstract**—In this paper we present a new approach for natural feature extraction using a laser scanner for the purpose of localization in outdoor environments. In semi-structured outdoor environments, naturally predominant features such as trees and edges are considered. The proposed method applies a batch processing which carries out feature extraction after measurements from a full scan are received. The algorithm consists of data segmentation and parameter acquisition. A modified Gauss-Newton method is proposed for fitting circle parameters iteratively. The natural features extracted through this approach are more robust than those obtained by existing methods. In order to reduce the estimation error caused by the linearization in the extended Kalman filtering (EKF), a particle filter is applied to realize the prediction and validation by integrating data from both the laser range sensor and encoder in outdoor environments. The proposed feature extraction and localization algorithms are verified in a real world experiment.

**Index Terms**— Feature extraction, Localization

## I. INTRODUCTION

Localization is a fundamental competence for autonomous mobile robot navigation systems. For almost every land navigation application we can always find an appropriate combination of dead reckoning sensors that can be used to obtain a reasonable prediction of the trajectory of the vehicle. Different sensors have different shortcomings. For outdoor applications, GPS is widely used but its accuracy can't be guaranteed when the satellite signal is blocked by trees or buildings. Ultrasonic sensors have been applied in indoor environment, but they are not adequate for outdoor situations due to the range limitation and bearing uncertainties. Stereo vision should be a promising technology, but its complexity and poor dynamic range make it unreliable for outdoor applications. Millimeter wave radar has enormous potential but it's very expensive. Recently, laser sensors have become one of the most attractive sensors for localization purpose due to their accuracy and low cost [1] [2]. Most laser sensors provide bearing and range information with sub degree resolution and an accuracy of the order of 1-10 cm and range detected can be up to 80 meters. In the past several years, the use of laser range measurement systems for autonomous navigation tasks has been on the rise [3], [4], [2], [5], [6]. However, the predominant domain of their use has been in indoor environments [3], [4], [5].

From a statistical point of view, robot localization is an on-line filtering problem: estimate the current state of the robot, given a sequence of observations [7], [8]. Many existing approaches rely on the EKF for robot state estimation [9] [10]. Although the EKF constitutes a powerful framework, its applicability is restricted by the required linearization of a nonlinear system and the assumption of Gaussian noises. This weakness may result that the EKF is very sensitive and gives inaccurate estimates. A popular algorithm which is able to deal with non-Gaussian distributions and nonlinear problems is the particle filter [11] [12].

In this paper, measurements from a laser sensor and encoders are combined to estimate the vehicle pose by using naturally occurring landmarks (trees or tree-like objects) in outdoor environments. A novel feature extraction approach is proposed. It applies a curve gradient model to detect large discontinuities of the environment in the process of data segmentation. A modified Gauss-Newton optimization approach is then proposed for acquisition of circle parameters. It is worth noting that there are few existing papers focused on outdoor feature extraction. The existing methods for tree detection can only give a rough estimate of circle parameters by a batch calculation [13] [14], therefore the results are very sensitive to measurements of large errors. The proposed method is more robust and more accurate in detecting circle features and acquiring the feature parameters. To test the effectiveness of this feature detection method, a particle filter based data association and localization framework is proposed. Under the model-based feature representation, the data association will be shown simplified. We only need to match the circle center coordinates using a validation gate. In order to deal with the nonlinear problem, a particle filter based algorithm is implemented for localization and data association.

The paper is outlined as follows. In Section 2, a new algorithm for natural landmark (tree or tree-like objects) extraction is proposed. Section 3 presents a particle filter based data association and localization implementation. Section 4 shows the experimental results in a real world outdoor environment. Finally, some conclusion is drawn in Section 5.

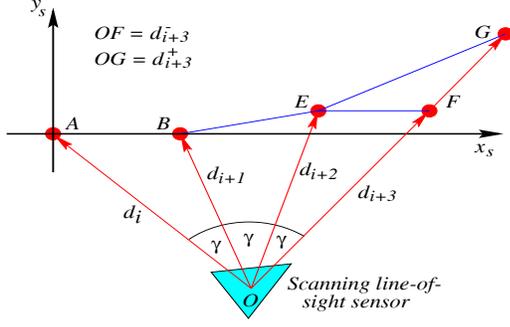


Fig. 1. Gradient model. The first three ranges of scan points are used to predict the next range value based on the assumption of similar gradient change in a smooth surface.

## II. FEATURE EXTRACTION

Our method to obtain reliable feature information for localization purpose is to use trees or tree-like objects which are the main class of features available in an outdoor environment. The center and radius of a circle feature can be used for navigation. We present a Unscented Kalman filter (UKF) and Gauss-Newton optimization based feature extraction method to obtain these natural features.

The essential components of this algorithm consist in two parts: the first is the segmentation of data from a scan and the other is the parameter acquisition.

During the data segmentation process, we apply a UKF to a curve gradient model to realize the data segmentation. Only large discontinuities of the measurement data are identified so as to divide the scan data into several groups. Then for each data group, a modified Gauss-Newton iteration optimization algorithm is applied to calculate the center and radius of each tree trunk. The estimated parameters are used for localization.

### A. Data Segmentation

Segmentation is a process of aiming to classify each scan data into several groups, each of which possibly associates with different structures of surrounding. The segmentation process is realized through the UKF [15]. At each time instant the range estimate is compared to the range measurement based on their statistics in order to decide if an edge has been detected. When the difference between the measured range and the predicted range is beyond certain threshold, we consider that an edge has been detected. This can be achieved by using a validation gate during the prediction process with the UKF.

Firstly, we shall establish a general curve gradient model for data segmentation. This gradient model is shown in Figure 1. The change in gradient between AB and BE is  $\Delta \left. \frac{dy_s}{dx_s} \right|_{x_s=B}$ , which can be determined as a function of  $d_i$ ,  $d_{i+1}$ ,  $d_{i+2}$  and  $\gamma$ .

$$\Delta \left. \frac{dy_s}{dx_s} \right|_B = \frac{num}{den} \quad (1)$$

$$num = (d_i d_{i+1} + d_{i+1} d_{i+2} - 2d_i d_{i+2} \cos \gamma) \sin \gamma$$

$$den = d_{i+1}^2 - d_i d_{i+1} \cos \gamma - d_{i+1} d_{i+2} \cos \gamma + d_i d_{i+2} \cos 2\gamma$$

For smooth surfaces, one can assume a similar change in gradient. The curve gradient model will be used to examine the feasibility of using the constant change in gradient estimate to detect large discontinuities (edges). From the estimate of  $\Delta \left. \frac{dy_s}{dx_s} \right|_B$ , a minimum value for the next range reading  $d_{i+3}^-$  is predicted, based on the negative spatial gradient change of  $-\Delta \left. \frac{dy_s}{dx_s} \right|_B$ , and a maximum value  $d_{i+3}^+$  is predicted based on the positive gradient change of  $+\Delta \left. \frac{dy_s}{dx_s} \right|_B$ . From the figure we can obtain:

$$d_{i+3}^- = \frac{(2d_i - d_{i+1})d_{i+2}}{d_i - 2d_{i+1} \cos \gamma + 2d_i \cos 2\gamma} \quad (2)$$

$$d_{i+3}^+ = d_{i+3}^- + \frac{2d_{i+2}(d_i^2 + d_{i+1}^2 - 2d_i d_{i+1} \cos \gamma)}{d_i - 2d_{i+1} \cos \gamma + 2d_i \cos 2\gamma} \times \frac{2d_i d_{i+2} \cos \gamma - d_i d_{i+1} - d_{i+1} d_{i+2}}{R + S + T + U} \quad (3)$$

where R, S, T and U are given by:

$$R = -2d_i d_{i+1}^2 - d_i^2 d_{i+2} - d_{i+1}^2 d_{i+2} + 2d_i^2 d_{i+1} \cos \gamma + 2d_{i+1}^3 \cos \gamma$$

$$S = d_i d_{i+1} d_{i+2} \cos \gamma - d_i d_{i+1}^2 \cos 2\gamma + d_i^2 d_{i+2} \cos 2\gamma$$

$$T = d_{i+1}^2 d_{i+2} \cos 2\gamma - d_i^2 d_{i+1} \cos 3\gamma - 2d_i d_{i+1} d_{i+2} \cos 3\gamma$$

$$U = d_i^2 d_{i+2} \cos 4\gamma$$

According to the two different changes in gradient for a smooth surface, there will be a minimum and a maximum predicted range values. If the observed range reading is not within  $3\sigma$  bounds of the minimum or maximum predicted value, the filter will be re-initialized and an edge is found.

1) *System Model*: To form a recursive prediction formula, we can rewrite (2) and (3) by defining:

$$x_1(k) = d_i; x_2(k) = d_{i+1}; x_3(k) = d_{i+2}$$

thus, we get the discrete system model for UKF:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3^-(k+1) \text{ or } \\ x_3^+(k+1) \end{bmatrix} = \begin{bmatrix} x_2(k) \\ x_3(k) \\ f^-(x_1(k), x_2(k), x_3(k)) \text{ or } \\ f^+(x_1(k), x_2(k), x_3(k)) \end{bmatrix} + v_1(k) \quad (4)$$

where  $f^-(x_1(k), x_2(k), x_3(k))$  or  $f^+(x_1(k), x_2(k), x_3(k))$  can be calculated from (2) and (3). Equation (4) represents a system model which will be used to predict the next range value from the sensor before the actual range measurement is recorded.

2) *Observation Model*: Since there is almost no angular uncertainty for the Sick sensor, we assume that its bearing is known precisely at each time instant, so we only consider the range measurement. Therefore, our observation model is:

$$z(k) = [0 \quad 0 \quad 1] \begin{bmatrix} x_2(k) \\ x_3(k) \\ f^-(x_1(k), x_2(k), x_3(k)) \text{ or } \\ f^+(x_1(k), x_2(k), x_3(k)) \end{bmatrix} + \omega_1(k) \quad (5)$$

where  $\omega_1(k)$  is a zero mean Gaussian noise with a known variance  $\sigma_r^2$ . By studying the statistics of the range measurement of the laser scanner, we found that the sensor noise follows approximately a Gaussian distribution with standard deviation ranging from 0.015 to 0.020 meter. We thus take the variance to be  $\sigma_r^2 = 0.00040$ . The UKF is used to realize the prediction and validation process.

3) *The Recursive UKF Process and Validation Gate*: From the state equation (4), it is observed that the curve gradient model is highly non-linear and its Jacobian matrix would be very complex, which would lead to a high computational demand, if the EKF is to be applied. Realizing this difficulty, we shall propose to use the UKF [15]. We apply the UKF to realize the prediction and validation process. But it is noted that during the UKF realization process for the curve gradient model, we choose  $\nu = \min\{f^-(x_1(k), x_2(k), x_3(k)) - z(k+1); f^+(x_1(k), x_2(k), x_3(k)) - z(k+1))\}$  to do the prediction and validation. For each instant, we need to calculate two predictions of the system variable using equation (4).

In order to identify if a measurement is associated with a new edge, certain criterion needs to be established. Use the innovation  $\nu(k+1)$  and the innovation variance  $s(k+1)$  to define:

$$d(k+1) = \nu(k+1)^T s^{-1}(k+1) \nu(k+1) \quad (6)$$

Note that since  $\nu$  is a Gaussian random variable,  $d$  is a random variable following the  $\chi^2$  distribution. The smaller the  $d(k+1)$ , the higher the probability that the measurement  $z(k+1)$  is obtained from the smooth surface. Thus, a validation gate,  $\delta$ , is used to decide whether the measurement  $z(k+1)$  is a close enough match to the predicted data point to continue the filter update. If the measurement is such that  $d(k+1) > \delta$ , a discontinuity is found. From the  $\chi^2$  distribution table, we know that if the observation is from the expected model surface, then  $d(k+1) < 6.63$  with a probability of 0.99. If a small  $\delta$  is selected, there will be more edges found. In the present application, the validation gate can be set to be large to tolerate those measurements that don't match the prediction very well, thus only the large discontinuities can be detected. Here we set  $\delta = 6.63$ . After the prediction and validation process, the large discontinuities can be found, thus, the data is divided into several groups.

After the data segmentation process, we need to decide if each segment of data is associated with an arc or not. In our algorithm, an accuracy limit or a maximum iterative number (if the accuracy limit can not be reached) is defined according to the experimental data to distinguish the arc data group from an "non-arc" data group. For an arc, we shall need to estimate its parameters (the center and the radius) of the arc so that future measurements of the arc may be used for robot navigation. In the following, the modified Gauss-Newton optimization method [16] is applied to obtain the parameters of each arc similar to [17].

## B. Parameter Acquisition

A circle can be defined by the equation

$$(x - x_0)^2 + (y - y_0)^2 = r^2 \quad (7)$$

where  $(x_0, y_0)$  and  $r$  are the center and the radius of the circle, respectively. For a circle fitting problem, the data set  $(x, y)$  is known and the circle parameters  $(x_0, y_0, r)$  need to be estimated. Assume that we have obtained  $M$  measurements  $(x_m, y_m)$ ,  $m = 1, 2, \dots, M$ , of the circle. Our objective is to find  $p = (x_0, y_0, r)$  that minimizes

$$E(p) = E(x_0, y_0, r) = \sum_{m=1}^M [(x_m - x_0)^2 + (y_m - y_0)^2 - r^2]^2 \quad (8)$$

This is equivalent to performing the nonlinear least-squares process using the equations

$$f_m(x_0, y_0, r) = (x_m - x_0)^2 + (y_m - y_0)^2 - r^2 = 0, \quad m = 1, 2, \dots, M \quad (9)$$

Equation (9) is not linear about the unknown parameters  $x_0$ ,  $y_0$ , and  $r$ , therefore it is a nonlinear least-squares problem. We propose to use the modified Gauss-Newton optimization (Levenberg-Marquart) method [16] to solve the problem.

In our case the Jacobian matrix for the modified Gaussian-Newton algorithm is

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial x_0} & \frac{\partial f_1}{\partial y_0} & \frac{\partial f_1}{\partial r} \\ \frac{\partial f_2}{\partial x_0} & \frac{\partial f_2}{\partial y_0} & \frac{\partial f_2}{\partial r} \\ \vdots & \vdots & \vdots \\ \frac{\partial f_M}{\partial x_0} & \frac{\partial f_M}{\partial y_0} & \frac{\partial f_M}{\partial r} \end{bmatrix} \quad (10)$$

Let  $\bar{f} = (f_1 \ f_2 \ \dots \ f_M)^T$  with  $f_m$  as defined in (9). At the  $k$ -th step, using the modified Gauss-Newton method to search the solution according to the following equation:

$$(A_k^T A_k + \lambda_k I) \Delta p_k = -A_k^T \bar{f}_k \quad (11)$$

where  $\Delta p_k = p_{k+1} - p_k$  and  $p_k$  is the estimate of  $p = [x_0 \ y_0 \ r]^T$  at the  $k$ -th iteration. We set the initial value  $\lambda_0 = 0.01$  and carry out the following iterations for calculating an suboptimal  $p$ :

Step 1: Calculate  $\Delta p_k$  using (11);

Step 2: Calculate the sum error  $E(p_k + \Delta p_k)$  by equation (8);

Step 3: Compare with the sum error of last step  $E(p_k)$ , if  $E(p_k + \Delta p_k) > E(p_k)$ , increase  $\lambda_k$  by a factor of 10, and go back to Step 1;

Step 4: If  $E(p_k + \Delta p_k) < E(p_k)$ , decrease  $\lambda_k$  by a factor of 10, update the trial solution, i.e. replace  $p_k$  by  $p_k + \Delta p_k$  and go back to Step 1 until the algorithm converges.

The convergence condition can be defined by the sum of the error square and the iterative step number. Thus the solution of  $x_0, y_0$ , and  $r$  will be acquired by the above multiple iterations.

A starting guess for these parameters are required. We use the first three points  $(x_i, y_i)$   $i = 1, 2, 3$  on the arc to compute an estimated initial value of  $(x_0, y_0, r)$ . The more accurate the initial value is, the faster the algorithm converges.

### III. DATA ASSOCIATION AND LOCALIZATION BY PARTICLE FILTER

#### A. Navigation Loop

The navigation loop is based on encoders and range/ bearing information provided by a laser sensor. The models for the process and observation are highly nonlinear. The encoders provide velocity and steering angle information that is used with a kinematic model of the vehicle to predict position and orientation. The prediction is updated with external range and bearing information provided by a laser sensor. A GPS is also used as a reference to judge if the pose estimation of the vehicle is accurate or not.

1) *Vehicle Model:* The vehicle named Cycab that is used in our research is a typical car-like vehicle; see Figure 2. We make the basic assumptions of planar motion, i.e. the vehicle is a rigid body and there is no slippage of tire. The status of the vehicle in motion can be described in the global coordinates by using the ‘‘bicycle model’’ [18]. To describe the vehicle motion, a global coordinate X-Y is fixed on the horizontal plane on which the vehicle moves. The motion status of the vehicle at an arbitrary moment can be described using the Bicycle Model as illustrated in Figure 3.

Reference point G is chosen at the center of gravity of the vehicle body. Its coordinates  $(x_v, y_v)$  represents the position of the vehicle; Vehicle Velocity  $v$  is defined at the reference point G; Heading Angle  $\theta_v$  is the angle from the X-axis to the longitudinal axis of the vehicle body; Side-slip Angle  $\beta$  is the angle from the longitudinal axis of the vehicle body to the direction of vehicle velocity,  $v$ ; Turning radius  $r$  is the distance between the reference point G and the Instant Rotating Center O; Front Wheel Velocity  $v_f$  is the velocity defined at the intersection of the mid-plane of the virtual front wheel and the front wheel axle; Rear Wheel Velocity  $v_r$  is the velocity defined at the intersection of the mid-plane of the virtual rear wheel and the rear wheel axle; Front wheel steering angle  $\delta_f$  is the angle from the longitudinal axis of the vehicle body to the direction of  $v_f$ . The vehicle model is as follows:



Fig. 2. The Cycab, a car-like vehicle in our experiment

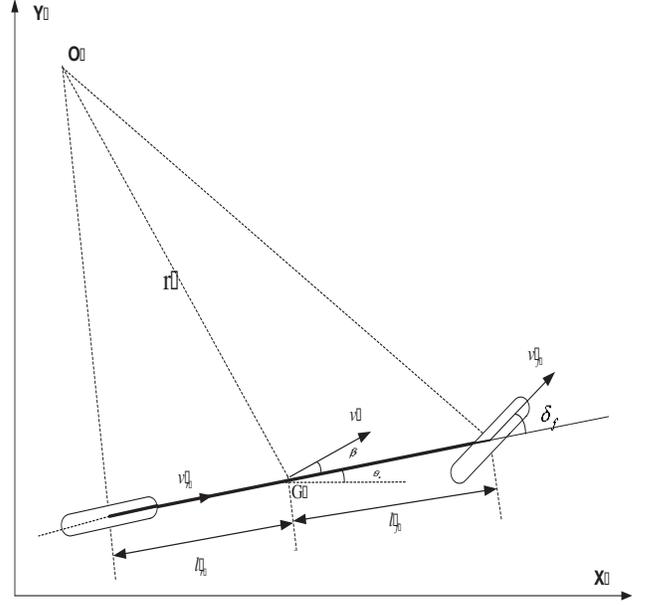


Fig. 3. The Cycab's Kinematic Model

$$\begin{bmatrix} x_v(k+1) \\ y_v(k+1) \\ \theta_v(k+1) \end{bmatrix} = \begin{bmatrix} x_v(k) + \Delta \cdot v \cos(\theta_v(k) + \beta) \\ y_v(k) + \Delta \cdot v \sin(\theta_v(k) + \beta) \\ \theta_v(k) + \Delta \cdot \frac{v}{r} \end{bmatrix} + \mathbf{v}_2(k) \quad (12)$$

where  $\Delta$  is the sampling interval;  $v$  and  $\beta$  are the control inputs that are computed from encoder increments during the interval  $\Delta$ . The above can be put into the general form:

$$\vec{x}(k+1) = \mathbf{f}(\vec{x}(k), u(k)) + \vec{v}_2(k) \quad (13)$$

2) *Sensor Model:* The state observation process can also be modelled in state-space notation by a non-linear vector function. Given the current vehicle position  $\vec{x}(k)$  and the position of an observed feature  $x_i(k)$ , the observation of range,  $z_r(k)$  and bearing,  $z_\theta(k)$ , can be modelled (similar to [1]) as follows:

$$\begin{aligned} \vec{z}_i(k) &= \begin{bmatrix} z_{ri}(k) \\ z_{\theta i}(k) \end{bmatrix} \\ &= \begin{bmatrix} \sqrt{(x_v(k) - x_i(k))^2 + (y_v(k) - y_i(k))^2} \\ \arctan\left(\frac{y_v(k) - y_i(k)}{x_v(k) - x_i(k)}\right) - \theta_v(k) \end{bmatrix} + \vec{w} \end{aligned}$$

where  $\vec{z}_i(k)$  is the measurement from the  $i$ th feature to the vehicle. It can be written in the following general form:

$$\vec{z}_i(k+1) = \mathbf{h}_i(\vec{x}(k+1), \vec{\eta}_i) + \vec{w}_i(k) \quad (14)$$

where  $\vec{\eta}_i = [x_i, y_i]^T$  denotes the  $i$ th feature state,  $\vec{x}(k+1)$  is the vehicle state and  $\vec{w}_i(k)$  is the measurement noise of the  $i$ th feature.

#### B. Particle Filter Based Implementation for Data Association and Localization

1) *Particle Filter Implementation:* In order to simplify the presentation, we rewrite the (13) and (14) as follows:

$$x_{k+1} = f(x_k, u_k, v_k) \quad (15)$$

$$z_{i,k+1} = h_i(x_{k+1}, \eta_i, \omega_{i,k}) \quad (16)$$

Based on the Bayesian estimation formula, the particle filter implementation algorithm for localization is derived as follows:

(1). Initialization: generate  $x_0^l \sim p_{x_0}$ ,  $l = 1, \dots, N$ . Each sample of the state vector is referred to as a particle. That is, start from a random measure with equal weight on each of the  $N$  sample values.

(2). Prediction: predict new particles, i.e.,

$$x_{k+1}^{(l)} = f(x_k^{(l)}, u_k, v_k^{(l)}), \quad l = 1, \dots, N$$

using different noise realizations of  $v_k$  for the particles. Compute the weights

$$w_{k+1}^{(l)} = p(z_{k+1} | x_{k+1}^{(l)})$$

and normalize the weight,

$$\tilde{w}_{k+1}^{(l)} = w_{k+1}^{(l)} / \sum_{j=1}^N w_{k+1}^{(j)}, \quad l = 1, \dots, N$$

In this step, we can also get the observation particles for the  $i$ th feature:

$$z_{i,k+1}^{(l)} = h_i(x_{k+1}^{(l)}, \eta_i) \quad (17)$$

$$\nu_{i,k+1}^{(l)} = z_{i,k+1} - z_{i,k+1}^{(l)} \quad (18)$$

Their mean is given by

$$\hat{\nu}_{i,k+1} = \sum_{l=1}^N \tilde{w}_{k+1}^{(l)} \nu_{i,k+1}^{(l)} \quad (19)$$

The variance of  $\hat{\nu}_{i,k+1}$  is:

$$\hat{\sigma}_{i,k+1} = \sum_{l=1}^N \tilde{w}_{k+1}^{(l)} (\nu_{i,k+1}^{(l)} - \hat{\nu}_{i,k+1})(\nu_{i,k+1}^{(l)} - \hat{\nu}_{i,k+1})^T \quad (20)$$

(3). Update: generate a new set  $\{x_{k+1}^{l*}\}_{l=1}^N$  by resampling with replacement  $N$  times from  $\{x_{k+1}^l\}_{l=1}^N$ , where

$$Pr(x_{k+1}^{l*} = x_{k+1}^{(l)}) = \tilde{w}_{k+1}^{(l)}, \quad l, l* = 1, \dots, N$$

Here, the mean of  $x_{k+1}^{(l*)}$  is:

$$\hat{x}_{k+1} = \sum_{l*=1}^N \frac{1}{N} x_{k+1}^{(l*)}$$

(4). Increase  $k$  and go back to step (2).

2) *Particle Filter Based Data Association*: Because the centers of the trees are applied as features for navigation, they are relatively stable and easy to match each other. We use the Nearest Neighbor data association method [14] here. The squared normalized innovation between the observation and the feature location,  $d_i^2$ , is then compared against a validation gate for the association being considered.

$$d_{i,k}^2 = \hat{\nu}_{i,k}^T \hat{\sigma}_{i,k}^{-1} \hat{\nu}_{i,k}$$

where  $\hat{\nu}_{i,k}$  and  $\hat{\sigma}_{i,k}$  can be calculated from (19) and (20) for each feature. The validated measurement nearest to the predicted measurement is used for updating the state of the vehicle.

## IV. EXPERIMENTAL RESULTS

The navigation algorithms presented were tested in the outdoor environment as shown in Figure 4. There are several tall trees and building walls and some bushes. The vehicle is Cycab, a car-like vehicle, as shown in the previous section. It is equipped with a laser range sensor—Sick LMS 200 and dead reckoning capabilities. There are four encoders fixed on the wheels of the vehicle and we can use this information to calculate the vehicle's position. A DGPS with up to 2cm accuracy is used as a reference to give the ground truth of the vehicle pose to get the estimation error. In the experimental environment, there are 8 tall trees and the vehicle moves along the path as shown in Figure 5 where the stars indicate the locations of the trees, the dashed blue line indicates the real pose of the vehicle and the real line means the estimated path using the proposed algorithms. Figure 6 shows a typical laser scanner frame. The dashed line box A indicates the region whose clearer view is shown in Figure 7. We can find that the trees are all detected. Compared to the ground truth, the radius and center coordinates are accurately extracted. In this case, we use the centers of the trees as features. It is more robust than only using raw sensor data [14].

Figure 8 shows the vehicle's position and orientation error in the prediction and their  $3\sigma$  bounds error. The  $3\sigma$  bounds of the error demonstrate that the estimation is satisfactory.



Fig. 4. The experimental environment (the whole scene).

## V. CONCLUSION

This paper presented a new feature detection algorithm for outdoor environment and the implementation of this approach using a particle filter based localization algorithm. The method has been realized in a real outdoor environment and satisfactory results have been obtained. It was demonstrated that the algorithm successfully extracted features of trees or tree-like objects which are robust for navigation because the center of a circle is relatively stable when vehicle moves. The  $3\sigma$  bounds of the estimation error also showed that the particle filter based localization performed well in real-time implementation. The main contribution of this paper is the natural feature detection algorithm and the particle filter based localization and their implementation.

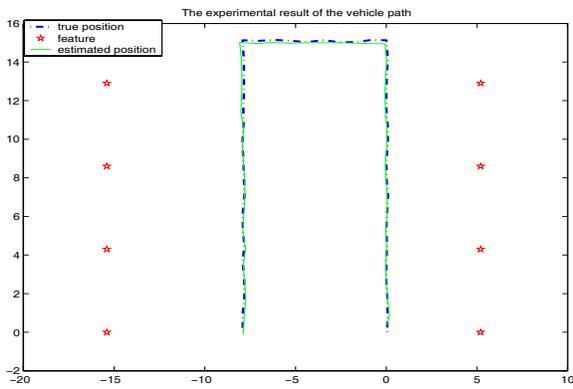


Fig. 5. The true vehicle path and the features' location

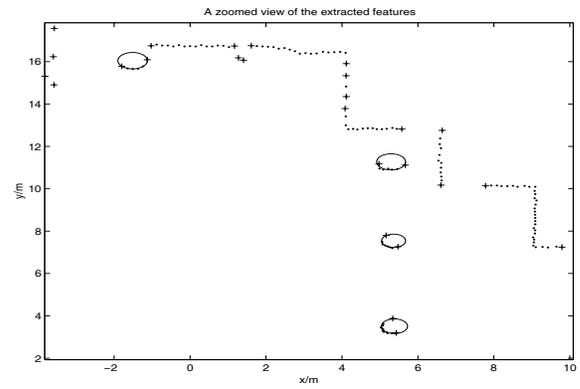


Fig. 7. A zoomed view of the region inside box A in Fig. 6.

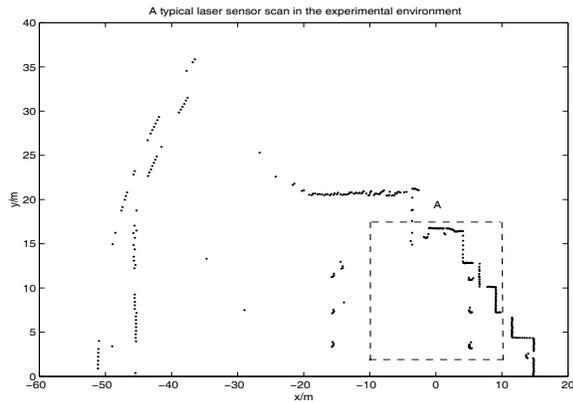


Fig. 6. Circles and edges extracted from data scanned using the proposed approach (the normal view).

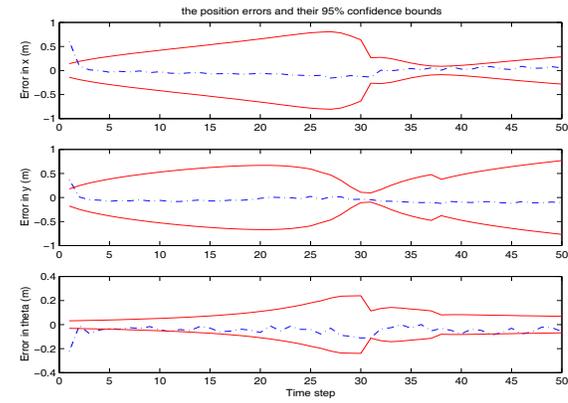


Fig. 8. The error and  $3\sigma$  error bounds of estimation error of the vehicle.

#### ACKNOWLEDGMENT

The authors would like to thank Professor D. W. Wang, Mr. Fan Tang, Mr. P. M. Tuan and Mr. C. B. Low for helpful discussion on data collection.

#### REFERENCES

- [1] J. Guivant, E. M. Nebot, and S. Baiker, "Localization and map building using laser range sensors in outdoor applications," *J. Robot. Syst.*, vol. 17, pp. 565–583, 2000.
- [2] E. M. Nebot, H. F. Durrant-Whyte, and S. Scheding, "Frequency domain modelling of aided gps for vehicle navigation systems," *Robotics and Autonomous Syst.*, vol. 25, pp. 73–82, 1998.
- [3] O. A. Kai, N. Tomatis, and R. Siegwart, "Multisensor on-the-fly localization using laser and vision," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Syst.(IROS)*, Takamatsu, Japan, October 2000, pp. 462–467.
- [4] B. Blanco and B. L. Boada, "Local mapping from on-line laser voronoi extraction," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Syst.(IROS)*, Takamatsu, Japan, October 2000, pp. 103–108.
- [5] O. Wijk and H. I. Christensen, "Localization and navigation of a mobile robot using natural point landmarks extracted from sonar data," *Robotics and Autonomous Syst.*, vol. 31, pp. 31–42, 2000.
- [6] S. Zhang, L. Xie, and M. D. Adams, "An efficient data association approach to simultaneous localization and map building," in *Proc. of IEEE Int. Conf. Robot. Automat.*, New Orleans, USA, April 2004, in press.
- [7] J. A. Castellanos, J. Neira, and J. D. Tardós, "Multisensor fusion for simultaneous localization and map building," *IEEE Trans. Robot. Automat.*, vol. 17, pp. 908–914, 2001.

- [8] J. E. Guivant, F. R. Masson, and E. M. Nebot, "Simultaneous localization and map building using natural features and absolute information," *Robotics and Autonomous Syst.*, vol. 40, pp. 79–90, 2002.
- [9] J. J. Leonard and H. F. Durrant-Whyte, "Mobile robot localisation by tracking geometric beacons," *IEEE Trans. Robot. Automat.*, vol. 7, pp. 376–382, 1991.
- [10] M. D. Adams and A. Kerstens, "Tracking naturally occurring indoor features in 2d and 3d with lidar range/amplitude data," *Int. J. Robot. Res.*, vol. 17, pp. 907–923, 1998.
- [11] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P. J. Nordlund, "Particle filters for positioning, navigation, and tracking," *IEEE Trans. Signal Processing*, vol. 50, pp. 425–436, 2002.
- [12] N. Vlassis, B. Terwijn, and B. Krose, "Auxiliary particle filter robot localization from high-dimensional sensor observations," in *Proc. IEEE Int. Conf. Robot. Automat.*, Washington, DC, May 2002, pp. 7–12.
- [13] J. Vandorpe, H. V. Brussel, and H. Xu, "Exact dynamic map building for a mobile robot using geometrical primitives produced by a 2d range finder," in *Proc. IEEE Int. Conf. Robot. Automat.*, Minnesota, April 1996, pp. 901–908.
- [14] T. Bailey, *Mobile Robot Localization and Mapping in Extensive Outdoor Environment*. Ph.D thesis, Sydney Univ., 2002.
- [15] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Trans. Automat. Contr.*, vol. 45, pp. 477–482, 2000.
- [16] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer Verlag, 1999.
- [17] S. Zhang, L. Xie, and M. D. Adams, "Geometrical feature extraction using 2d range scanner," in *Proc. of the Fourth Int. Conf. Contr. and Automat.*, Montreal, Canada, June 2003, pp. 901–905.
- [18] D. Wang, E. K. Tay, and M. Zribi, "Modeling of an agv for handling heavy containers," in *Proc. of the 3rd Int. Conf. advanced mechatronics*, Japan, August 1998, pp. 67–72.