2005 IEEE/RSJ International Conference on Intelligent Robots and Systems

Entropy Based Feature Selection Scheme for Real Time Simultaneous Localization and Map Building

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

***Abstract*— We propose a novel entropy-based method for feature selection in order to reduce the computational burden for real time simultaneous localization and map building (SLAM) for mobile robot navigation. Our approach is based on information (entropy) theory together with a data association method to initialize new features into the map, match measurements to the map features, and remove out-of-date features. The selected features are optimum in the sense that fusion of measurements from those features with existing information would yield the most entropy reduction in estimating the robot location and the map features' locations. Our method has the advantage of selecting a suitable number of features by considering the computational constraint in real time implementations. Simulation results show that the proposed entropy based feature selection strategy is effective in dealing with the map scaling problem in SLAM.**

***Index Terms*— Entropy, Feature Selection, SLAM**

I. Introduction

The problem of SLAM has attracted immense attention in the mobile robotics literature. The objective of SLAM is to use the information obtained by sensors mounted on a vehicle to build and update a map of the environment and simultaneously compute the vehicle location in that map. SLAM has been a central research topic in the field of mobile robotics due to its theoretical challenges and critical importance for many different types of robot applications [1]. A key stumbling block in the development and implementation of new methods for SLAM has been the map scaling problem—the increase of computational complexity with the size of the operating environment of the mobile robot.

It is well known that the complexity of SLAM algorithms can be reduced to $\sim O(N^2)$ [2], where N is the number of landmarks in the map. If the SLAM period is long, the number of the landmarks will increase and at last the computer resources will not be sufficient to update the map in real time. Real-time performance becomes impossible for environments with more than a few hundred features. There have been some approaches towards solving the map scaling problem. Leonard et al presented a method which splits the map into multiple globally-referenced submaps, each with its own vehicle track, and maintains all correlations within a submap [3]. The motivation is to achieve good performance by computing multiple partial solutions in parallel, and to avoid the computational burden that is entailed by computing one complete solution. They also present an approximation technique to address the update of covariances in the transition between the maps. However, a heuristic algorithm is needed to propagate the vehicle estimates between submaps. Although they present impressive experimental results, there is no proof of the consistency of the approach or estimation of the conservatism of the covariance over-bounding strategy. Newman analyzed the convergence properties of estimation error covariance matrices in SLAM and develops techniques for the scaling problem based on the use of relative maps [4]. It can be shown that the scaling and computational costs vary between being linear in the number of beacons and the same as the optimal solution. However, there are two difficulties with this method. One is that beacon estimates are not expressed in the same consistent coordinate frame and must be converted using projection operators [4]. The other problem is that the map cannot be used to directly update the vehicle. Simon Julier and Uhlmann introduced a covariance intersection (CI) for state estimation in SLAM [5]. The computational costs are constant time and the scaling is linear in the number of beacons [6]. However, SLAM implemented with CI may not provide an effective solution because the error bounds can be too conservative. Guivant and Nebot presented a compressed filter to store and maintain all the information gathered in a local area with a cost proportional to the square of the number of the landmarks in this area [1]. They have shown methods to reduce the number of times that the full $\sim O(N^2)$ update must be performed.

In this paper, we present an entropy based feature selection strategy to reduce the computational cost during SLAM. Practically, information methods provide a natural way of mixing measures of continuous information gain with discrete information gain. By using entropy information, we get that the next selection step is performed so as to maximize the robot's information about its location and all features' locations in the map. The algorithm can select available computationally resourceful features. Feature selection is formulated as an optimization problem that maximizes the amount of the information acquired. The information gained by observing an environmental feature must counteract the rise in uncertainty that results from the motion of the vehicle.

The paper is organized as follows. Section 2 is devoted to the entropy-based feature selection formulation of SLAM. Section 3 presents a method to solve the formulated problem and demonstrates how to combine it with the data association in SLAM so as to obtain a real-time SLAM solution. Section 4 shows simulation and experimental results. Some conclusions are drawn in Section 5.

II. Entropy-based Feature Selection for SLAM

A. Stochastic Mapping

We firstly review the theory of stochastic mapping by using standard Bayesian estimation theory. It is simply a special way of organizing the states in an Extended Kalman Filter (EKF) for the purpose of feature based SLAM. Assume that the kinematic model of the vehicle is as follows:

$$\mathbf{x}_v(k+1) = \mathbf{f}_v(\mathbf{x}_v(k), \mathbf{u}(k)) + \mathbf{v}_v(k) \tag{1}$$

where

$$\mathbf{x}_v(k+1) = \begin{bmatrix} x_v(k+1) \\ y_v(k+1) \\ \theta_v(k+1) \end{bmatrix}$$

with x_v and y_v being the coordinates of the vehicle and θ_v the heading angle. $\mathbf{f}_v$ defines the kinematic model of the vehicle with $\mathbf{u}(k)$ being the vector of control inputs. $\mathbf{v}_v(k)$ is a white noise with zero mean and covariance $\mathbf{Q}_v$.

The location of the ith landmark is denoted as p_i. The state equation for the ith landmark is

$$\mathbf{p}_i(k+1) = \mathbf{p}_i(k), \quad i = 1, 2, \cdots, N$$

since landmarks are assumed stationary, where the number of landmarks in the environment is assumed to be N. The augmented state equation containing both the state of the vehicle and the states of all landmarks is denoted

$$\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k)) + \mathbf{v}(k) \tag{2}$$

where

$$\mathbf{f}(\mathbf{x}(k), \mathbf{u}(k)) = \begin{bmatrix} \mathbf{f}_v(\mathbf{x}_v(k), \mathbf{u}(k)) \\ \mathbf{p}_1(k) \\ \vdots \\ \mathbf{p}_N(k) \end{bmatrix}$$

and

$$\mathbf{v}(k) = [\, \mathbf{v}_v^T(k) \;\; 0 \;\; \cdots \;\; 0 \,]^T$$

The vehicle is equipped with a sensor that can obtain observations of the relative location of landmarks with respect to the vehicle. The observation model for the ith landmark is written in the form

$$\mathbf{z}_i(k) = \mathbf{H}_i(\mathbf{x}(k), \mathbf{p}_i(k)) + \mathbf{w}_i(k) \tag{3}$$

where $\mathbf{w}_i(k)$ is a white noise with zero mean and variance σ_r. The output function $\mathbf{H}_i(\cdot, \cdot)$ relates the output of the sensor to the state vector when observing the ith landmark. The a posteriori PDF of $\mathbf{x}_{k+1}$, given a set of measurements $\mathbf{Z}_{k+1} = \{\mathbf{z}_{k+1}, \cdots, \mathbf{z}_1\}$, can be found by the Bayes Rule as:

$$p(\mathbf{x}_{k+1} \mid \mathbf{Z}_{k+1}) = \frac{p(\mathbf{z}_{k+1} \mid \mathbf{x}_{k+1}) p(\mathbf{x}_{k+1} \mid \mathbf{Z}_k)}{p(\mathbf{z}_{k+1} \mid \mathbf{Z}_k)}. \tag{4}$$

The distribution $p(\mathbf{z}_{k+1} \mid \mathbf{x}_{k+1})$ is defined as the likelihood function by the Likelihood Principle. By knowing $p(\mathbf{x}_{k+1} \mid \mathbf{Z}_{k+1})$, we can form an estimate $\hat{\mathbf{x}}_{k+1}$ of the state. An EKF is an approximation of Bayesian formula. It is a computationaly efficient estimator for the states of a given nonlinear dynamic system and it assumes that the noise processes are well modelled by Gaussian noise and that the errors due to linearization of the nonlinear system are small. That is, the EKF for a system provides an estimate of both the state of the system $\hat{\mathbf{x}}$ as well as an estimate of the covariance of the estimated state $\mathbf{P}$. The covariance of the estimated state provides an estimate of the confidence in the estimate $\hat{\mathbf{x}}$. The estimation error covariance $\mathbf{P}$ of the system takes the form of:

$$\mathbf{P}_{k|k} = \begin{bmatrix} \mathbf{P}_{rr} & \mathbf{P}_{r1} & \cdots & \mathbf{P}_{rN} \\ \mathbf{P}_{1r} & \mathbf{P}_{11} & \cdots & \mathbf{P}_{1N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{Nr} & \mathbf{P}_{N1} & \cdots & \mathbf{P}_{NN} \end{bmatrix}_{k|k} . \tag{5}$$

The sub-matrices $\mathbf{P}_{rr}$, $\mathbf{P}_{ri}$ and $\mathbf{P}_{ii}$ are the vehicle to vehicle, vehicle to feature, and feature to feature covariances, respectively. This form is significant as it allows us to separate the uncertainty associated with the robot as well as each individual feature. This separation will be used in obtaining a measure of the information in our system. Thus, the robot and the map are represented by a single state vector, $\mathbf{x}$, with an associated estimation error covariance $\mathbf{P}$ at each time step. As new features are added, $\mathbf{x}$ and $\mathbf{P}$ increase in size.

B. Problem Formulation

In this section, we model the entropy-based feature selection starting with standard Bayesian estimation theory. For simplicity, we rewrite (4) in another form:

$$p(\mathbf{x} \mid \mathbf{z}) = \frac{p(\mathbf{z} \mid \mathbf{x}) p(\mathbf{x})}{p(\mathbf{z})}. \tag{6}$$

The system state $\mathbf{x}$ is the parameter to be estimated. The probability that the robot and the features have a pose $\mathbf{x}$ is given by $p(\mathbf{x})$. In Equation (6), $p(\mathbf{x} \mid \mathbf{z})$ is computed from sensor model and the robot's map of the environment. It may depend on two factors. The first is the noise corruption to the signal observed by the sensor. The second is the estimated system state $\mathbf{x}$. It includes the robot's location and features' locations estimated at each time step. These two factors bring the uncertainty to the sensor model.

Let $\mathbf{Z}$ be all measurements that have already been used in the inference of the current belief $p(\mathbf{x} \mid \mathbf{Z})$ of the robot and features' locations. Let $\mathbf{z}_i$ be the observation of an additional feature that hasn't been used so far, that is, $\mathbf{z}_i$ does not belong to $\mathbf{Z}$. The fusion of the additional feature $\mathbf{z}_i$ and the current system state belief $p(\mathbf{x} \mid \mathbf{Z})$ yields the next system state belief $p(\mathbf{x} \mid \mathbf{z}_i, \mathbf{Z})$ as:

$$p(\mathbf{x} \mid \mathbf{z}_i, \mathbf{Z}) = C p(\mathbf{z}_i \mid \mathbf{x}, \mathbf{Z}) p(\mathbf{x} \mid \mathbf{Z}) \tag{7}$$

where C is a normalization constant. We assume that $\mathbf{Z}$ and $\mathbf{z}_i$ are independent conditioned on $\mathbf{x}$, that is, $p(\mathbf{z}_i \mid \mathbf{x}, \mathbf{Z}) = p(\mathbf{z}_i \mid \mathbf{x})$, then Equation (7) becomes:

$$p(\mathbf{x} \mid \mathbf{z}_i, \mathbf{Z}) = Cp(\mathbf{z}_i \mid \mathbf{x})p(\mathbf{x} \mid \mathbf{Z}). \tag{8}$$

According to information theory [7], the uncertainty of an arbitrary continuous distribution $p(x)$ can be described by its entropy:

$$H(p_x) = -\int p(\mathbf{x}) \log(p(\mathbf{x}))d\mathbf{x}. \tag{9}$$

This measure can be considered as the "purity" of the probability distribution. If the distribution is highly focused at a single pose $\mathbf{x}$, then the entropy will be low. If the distribution is spread over a wide space, the entropy will be high. According to Equation (9), the effect that a particular set of sensing data has on the robot's and features' belief in their positions can be measured in this way.

$$H(p_{x|Z}) = -\int p(\mathbf{x} \mid Z) \log(p(\mathbf{x} \mid Z))d\mathbf{x}. \tag{10}$$

Combining Equation (6) and Equation (10), we give the entropy of the posterior distribution after obtaining the measurements:

$$H(p_{x|Z}) = -\int \frac{p(\mathbf{z} \mid \mathbf{x})p(\mathbf{x})}{p(\mathbf{z})} \log(\frac{p(\mathbf{z} \mid \mathbf{x})p(\mathbf{x})}{p(\mathbf{z})})d\mathbf{x}. \tag{11}$$

This equation gives the entropy of the posterior distribution before additional features are added. Recall that $p(\mathbf{x})$ is the prior position distribution and $p(\mathbf{z} \mid \mathbf{x})$ is the probability of the sensor measurement conditioned on the position, computed from the sensor model and the environmental map.

If we use an additional feature to update the system state, the uncertainty of the next system state $p(\mathbf{x} \mid z_i, Z)$ is a function of both $\mathbf{Z}$ and the additional feature measurement z_i, and can be described by the entropy $H(p_{x|z_i,Z})$ as:

$$H(p_{x|\mathbf{z}_i,Z}) = -\int p(\mathbf{x} \mid \mathbf{z}_i, \mathbf{Z}) \log(p(\mathbf{x} \mid z_i, \mathbf{Z}))d\mathbf{x}. \tag{12}$$

We want to find the additional features that can give more information gain if they are indeed fused with the existing features' and robot's pose distribution.

$$i = \arg\max_{i \in I}(H(p_{x|Z}) - H(p_{x|z_i,Z})).$$

where I is the information of the new features.

III. Entropy Based Feature Selection Strategy

The essence of our model in the last section is to determine which features should be selected to update the system state in SLAM so that they maximize the total knowledge (i.e., the information) about the system in the presence of measurement and navigational uncertainty. By choosing features, we mean that some "good" features are chosen so as to maximize the information gain about the robot's location and the locations of all the features (the map). The proposed approach for the selection of a particular landmark is based on localization and mapping information offered by a particular landmark from a given time step. It takes into account the uncertainty in the vehicle pose estimate and the existing features' positions in the map while computing the information content of the landmark.

The concept of entropy is employed to facilitate landmark augmentation. As stated in the last section, the entropy is minimum when the information is maximum. It is conventional to seek minimal entropy when actually maximum information is sought. A mathematical expression for the entropy of a Gaussian distribution is to be used in the following. For an n dimensional state vector $\mathbf{x}_{k+1}$ conditioned on an observation vector $\mathbf{Z}_{k+1}$, the posterior entropy can be obtained as:

$$H_{k+1|k+1} = E\{-\ln p\ (\mathbf{x}_{k+1} \mid \mathbf{Z}_{k+1})\}. \tag{13}$$

For a Gaussian distributed system state, we can calculate that the value of the above equation [7] as

$$H_{k+1|k+1} = 0.5 \ln[(2\pi e)^n \mid \mathbf{P}_{k+1|k+1} \mid]. \tag{14}$$

Therefore, for a Gaussian distribution vector all that is required to compute its entropy is the dimension n and covariance $\mathbf{P}$. The posterior and prior information metrics can then be defined as

$$im_{k+1|k+1} = -H_{k+1|k+1} = -0.5 \ln[(2\pi e)^n \mid \mathbf{P}_{k+1|k+1} \mid], \tag{15}$$

$$im_{k+1|k} = -0.5 \ln[(2\pi e)^n \mid \mathbf{P}_{k+1|k} \mid]. \tag{16}$$

Thus, we can calculate the information difference as follows:

$$\triangle i = im_{k+1|k+1} - im_{k+1|k}. \tag{17}$$

In our application, we can get Equation (18) below according to the above formula. When observations are recorded at some time step, we will perform the following actions to select the useful features which can provide us more information for the vehicle's pose and map features' positions. We firstly calculate the individual update covariance $\mathbf{P}_{k+1,i|k+1,i}$. This means we use only one feature at time step $k+1$ to update the system to get a covariance $\mathbf{P}_{k+1,i|k+1,i}$. The same action is performed to all observations at the same time step. Thus, we obtain the possible contribution for localization and mapping for each observation.

$$im_{k+1|k+1,i} = 0.5 \ln[(2\pi e)^n \mid \mathbf{P}_{k+1,i|k+1,i} \mid] \tag{18}$$

Then we will calculate the information difference for each feature in the map at time step k+1 as follows:

$$\triangle i = im_{k+1|k+1,i} - im_{k+1|k} \tag{19}$$

Those features that have large values of $\triangle i$ will be selected. In the entropy calculation, in order to avoid errors if the covariance tend to infinity in some cases, we can use another suitable information-based cost function proposed in [8].

$$C(P) = \pi \cdot \prod_j \sqrt{\lambda_j(\mathbf{P}_{vv})} + \pi \cdot \sum_{i=1}^{nf} \prod_j \sqrt{\lambda_j(\mathbf{P}_{ii})} \quad (20)$$

$$(21)$$

$$= \pi \cdot \sqrt{det(\mathbf{P}_{vv})} + \pi \cdot \sum_{i=1}^{nf} \sqrt{det(\mathbf{P}_{ii})} \quad (22)$$

where $\lambda_j(\cdot)$ is the jth eigenvalue of $\mathbf{P}_{vv}$ or $\mathbf{P}_{ii}$ and n_f is the number of features.

Now we focus on the map management problem. As we know, once features have been detected, they must be matched against known landmarks in the environment. Data association is firstly performed between the observed features and the features currently in the map. This step is one of the most crucial steps in the mapping process. When data is received from a sensor there is a possibility that it may in fact be a spurious measurement. Some spurious measurements can be eliminated by the development of appropriate feature extraction approaches [2]. Regardless of the care that is taken in designing the feature extractor, some spurious measurements may still be passed to the localization and mapping algorithm and it is important to have a mechanism for rejecting these. Here we adopt the strategy as in [9]. But the NN data association is replaced by a more effective matching algorithm in [10] in order to reduce incorrect matchings. Tentative features that are not reobserved are removed from the list after a fixed time interval has elapsed. In detail, the data association and map management can be summarized as follows:

- Matching: The features and the observations are matched by the IHGR method proposed in [10], and measurements that have not been matched are stored.
- Feature selection: Select features according to their entropy values. The number of selected features can be decided by an appropriate compromise between quality estimation of vehicle pose and available computational memory.
- Updating the system state: Update the vehicle and features' positions with the selected features with the EKF.
- Initializing new features: New features which are not matched with the features stored in the map by the IHGR algorithm are initialized.
- Removing old features: Out-of-date features are deleted.

IV. Simulation and Experimental Results

A. Simulation Results

The first test environment is established by randomly generating some features and assuming that the vehicle's trajectory is a circle whose radius is 62 meters. The robot moves at a constant speed and the heading angle changes 1 degree at each sampling instant. The environment has 160 features which are randomly distributed in a region of 150 meters by 150 meters. When the vehicle moves, some of the features are observed. We assume that the features' positions are unknown

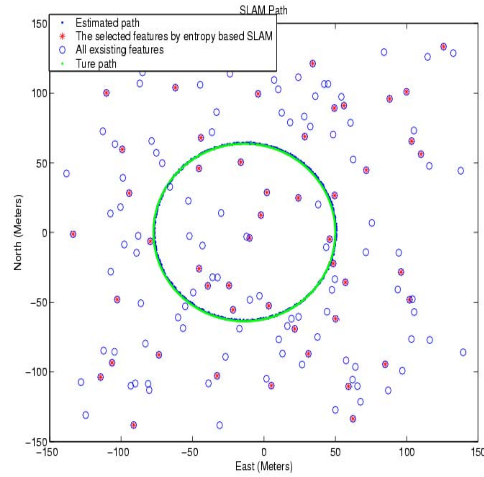

Fig. 1. The mapping and vehicle path when using entropy based feature selection in the SLAM process.

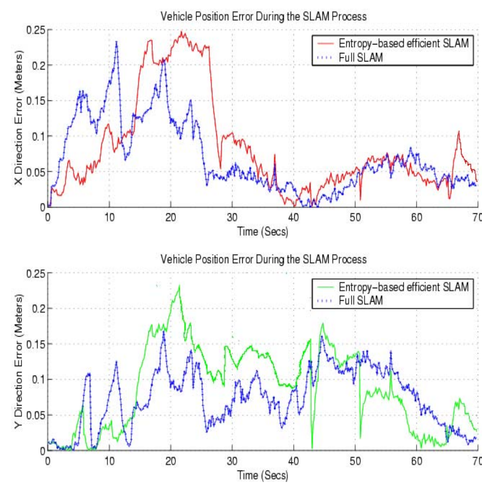

Fig. 2. The vehicle's position errors (between the ground truth (GPS reading) and the estimated position) comparison in the global coordinate between the entropy based feature selection in the SLAM process and full SLAM.

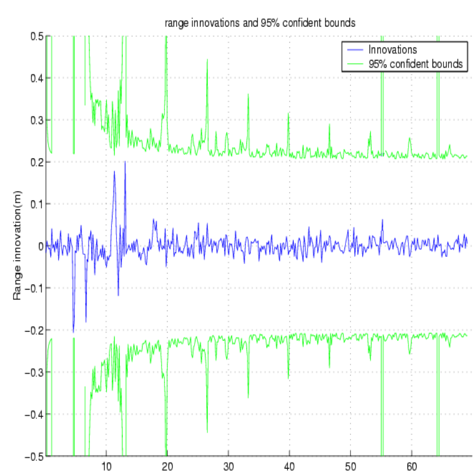

Fig. 3. The range innovation and its 2σ confidence bounds during the SLAM process with IHGR data association.

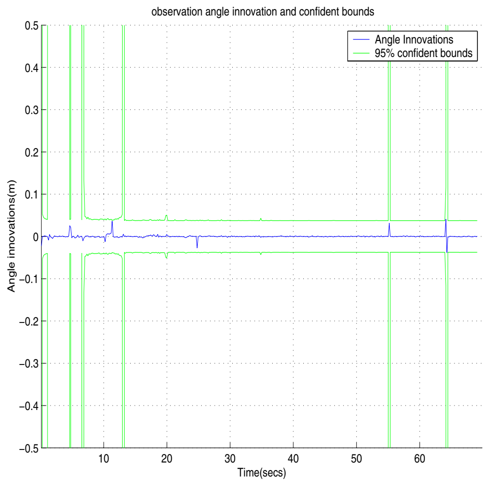

Fig. 4. The angle innovation and its 2σ confidence bounds during the SLAM process with IHGR data association.

which is the case in SLAM and match the features with the observations by using the IHGR algorithms [10] [11].

By applying the entropy based feature selection scheme proposed in the previous section, we implemented SLAM and the results are shown in Figure 1. Here, the features' number is selected as 10 after a fixed time interval (20 laser scans). It shows the SLAM results that include the vehicle's estimated trajectory and the features' position. The circles correspond to the features existing in the full SLAM implementation and the stars mean the features selected during the efficient SLAM process by using the entropy based feature selection scheme presented here. In fact, the vehicle's true path is almost overlapped with the estimated one during the SLAM process. We also show the global error between the estimated vehicle position and the ground truth which can be seen in Figure 2. In Figure 2, we can see that the comparison between the proposed SLAM global error and the full SLAM global error when we use all the obtained features without the proposed feature selection scheme. From the figure, we can see that the feature selection algorithm gives good results compared to the full (using all features) SLAM and with a lower computational cost. In full SLAM, the whole SLAM process where the vehicle travelled several hundred meters took 91.5482 seconds while the proposed algorithm took 80.7560 seconds on the same computer (the algorithms were run on a Pentium IV PC, 1.7GHz).

B. Experimental Results

In this subsection, a practical implementation of the proposed entropy-based feature selection scheme in an outdoor environment in Nanyang Technological University is presented. The testing site is a road around Canteen 3 and Hall 7 at Nanyang Technological University. Many trees and pillars exist on both sides of the road. Figure 5 shows the beginning part of the experimental environment. The vehicle is equipped with a GPS, a laser sensor and wheel encoders. A kinematic GPS system of 2 cm accuracy is used to evaluate the ground truth. Thus, the true navigation map is available for comparison purposes. Wheel encoders give an odometric measurement of the vehicle location. The dead reckoning sensors and laser range sensor are combined together to predict the vehicle's trajectory using the extended Kalman filter and to build the map at the same time. In this experiment, the features used are natural landmarks (mainly trees and pillars on the road) that are extracted by applying the feature detection algorithm in [12]. Feature detection was done by using a curve gradient model for data segmentation and a Gauss-Newton optimization method to obtain the most likely centers of tree trunks.

Figure 6 shows the real map of the experimental environment where the mobile robot travelled. The road is hundreds of meters long. Figure 7 shows the comparison of the estimated trajectory of the vehicle and the true path. It also shows the features extracted from the raw laser sensor data which is represented by the circles and the selected features which are applied for the SLAM update, and are marked by stars. Figures 8 and 9 give the validated range and bearing innovation sequences along with the 3σ bounds. It is evident that the innovations of range and bearing are well within the bounds which shows that the estimate is consistent.

Fig. 5. Part of the experimental site (we started from this point).

Fig. 6. The real map of the vehicle trajectory at Nanyang Technological University.

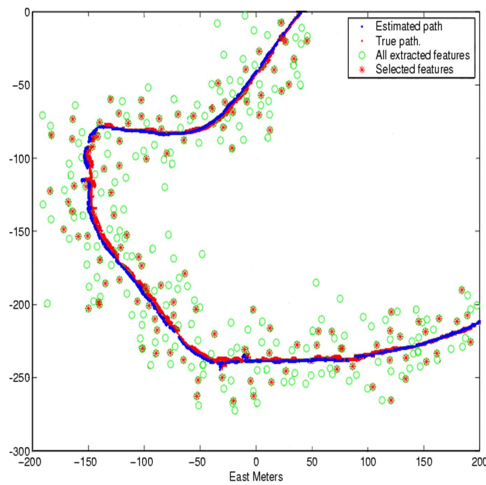

Fig. 7. The experimental results. the figure shows the vehicle's estimation path, the features extracted (circles) and the selected features (stars) by using the entropy based feature selection algorithm during the SLAM process.

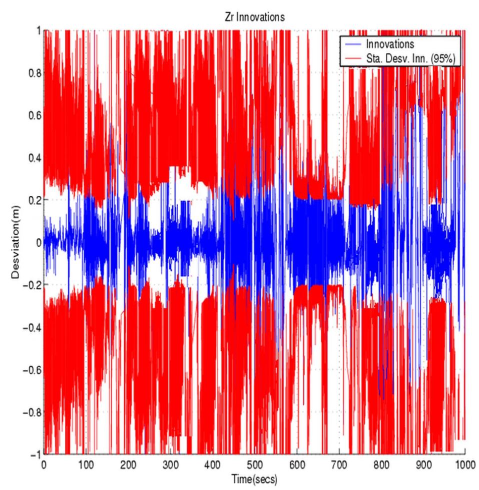

Fig. 8. The range innovation and its 3σ confidence bounds during the SLAM process with the entropy based SLAM.

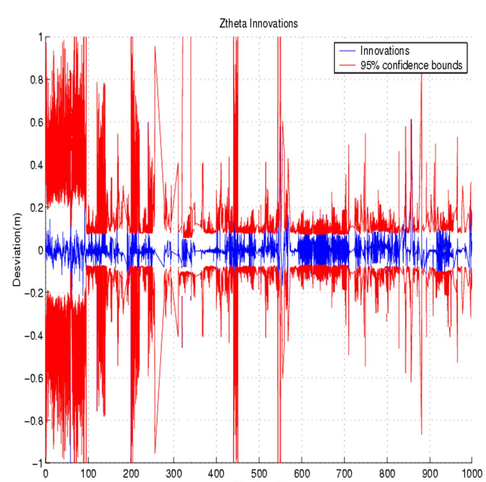

Fig. 9. The observation angle innovation and its 3σ confidence bounds during the SLAM process with the entropy based SLAM.

V. Conclusions

This paper considered the problem of selecting features for feature-based SLAM with autonomous outdoor vehicles. A novel feature selection strategy was incorporated within a stochastic mapping algorithm and tested via simulations and real data in an outdoor environment. We introduced a method for performing SLAM in a priori unknown environments with large number of features. The method is based on choosing features that, given the current knowledge, would maximize the information gain in the estimation of the vehicle's pose and the features' position. We can select the number of features to update the SLAM state according to the availability of computational memory. This approach can easily be implemented as an extra step in a stochastic mapping algorithm for SLAM. The validity and usefulness of the approach has been verified by simulations and experiments.

Acknowledgments

The authors would like to thank Associate Professor Wang Dan Wei and his research group for their kind help.

References

[1] J. E. Guivant and E. M. Nebot, "Optimization of the simultaneous localization and map-building algorithm for real time implementation," *IEEE Trans. Robot. Automat.*, vol. 17, pp. 242–257, 2001.

[2] G. Dissanayake, H. F. Durrant-Whyte, and T. Bailey., "A computationally efficient solution to the simultaneous localisation and map building (slam) problem," in *Proceedings of the IEEE International Conference on Robotics and Automation*, San Francisco, USA, April 2000, pp. 1009–1014.

[3] J. J. Leonard and H. J. S. Feder, "Decoupled stochastic mapping," *IEEE Journal of Oceanic Engineering*, vol. 26, pp. 561–571, 2001.

[4] P. M. Newman, *On the Structure and Solution of the Simultaneous Localization and Mapping Problem*. Australian Center for Field robotics, Univ. of Sydney, 1999.

[5] J. K. Uhlmann, *Dynamic map building and Localization for autonomous vehicles*. Univ. of Oxford, 1995.

[6] S. J. Julier and J. K. Ulmann, "simultaneous localization and map building using split covariance intersection," in *Proc. of the 2001 IEEE/RSJ Int. Conf. Intellegent Robots and Systems*, Hawaii, USA, November 2001, pp. 1257–1262.

[7] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 1991.

[8] H. Feder, J. Leonard, and C. Smith, "Adaptive mobile robot navigation and mapping," vol. 18, pp. 650–668, 1999.

[9] S. B. Williams, *Efficient Solutions to Autonomous Mapping and Navigation Problems*. Ph.D thesis, University of Sydney, 2001.

[10] S. Zhang, L. Xie, and M. D. Adams, "An efficient data association approach to simultaneous localization and map building," *International Journal of Robotics Research*, vol. 24, pp. 49–60, 2005.

[11] ——, "An efficient data association approach to simultaneous localization and map building," in *Proc. of IEEE Int. Conf. Robot. Automat.*, New Orleans, USA, April 2004, pp. 1493–1498.

[12] ——, "Gradient model based feature extraction for simultaneous localization and mapping in outdoor applications," in *Proceedings of the Eighth International Conference on Control, Automation, Robotics and Vision (ICARCV 2004)*, Kunming, China, December 2004.