Proceedings of the 5th World Congress on Intelligent Control
and Automation, June 15-19, 2004, Hangzhou, P.R. China

第五届全球智能控制与自动化大会
2004 年 6 月 15 - 19 日，中国杭州

Robust Mobile Robot On-the-Fly Localization: using Geometrical Feature Matching Method*

Yingjie Sun and Qixin Cao
Research Institute of Robotics
Shanghai Jiao Tong University
Shanghai, 20030, China
{yjsun & qxcao}@sjtu.edu.cn

Jay Lee
Intelligent Maintenance Systems
University of Wisconsin
Milwaukee, 53201,USA
jaylee@uwm.edu

***Abstract* - Self localization is an essential component of any autonomous mobile robot system. This paper presents a fast and robust feature-based approach to mobile robot on-the-fly global localization in dynamic environments. A robust feature extraction and correspondence algorithm was introduced to extract the landmarks of the environment and to triangulate the robot's positions. The approach take into account the hard real time constraint and the partial observability of the environment. On-line localization is performed by matching candidate landmarks from the robot's lcoal view to the tracked landmarks. Experimental validation in the context of Middle Size League RoboCup was carried out in a real environment with the mobile robot Jiaolong which equips a Laser Range Finder (LRF).**

Index Terms - On-the-fly localization; Feature extraction and correspondence; Geometrical feature matching; Landmark Detection, Laser Range Finder.

I. INTRODUCTION

Localization is an essential area of mobile robotics that has received increased attention over the past decade [1]. Localization is the process of determining the robot's position within a given environment, and it is a procedure that takes a set of sensor readings and/or *a priori* map as input and gives an estimate of the robot's pose. Navigation is a fundamental problem for autonomous mobile robots and it relies highly on robust localization methods [2]. The use of a particular kind of sensor usually affects the design choices for localization method. There are many successful localization methods that are able to determine a robot's position relative to a map using odometer, sonar, Laser Range Finder and camera [2, 3].

Localization methods are discriminated as relative and global localization. Relative localization method generally employs proprioceptive sensors (e.g. odometer, inertial sensor and gyro) to estimate the robot motion, and to update the robot's location. Dead reckoning based techniques are impractical because the systematic and nonsystematic measurement errors grow without bound due to the slippage and collision. The problem of global localization is to estimate the correct pose of a mobile robot with respect to some global reference frame from little of no *a priori* position and orientation.

Geometrical approaches require a comparison between the local geometry model extracted from the sensor information with *a priori* environment. Feature based localization is an extensively used geometrical method [4, 5]. The process of feature extraction and correspondence must be executed on line while the robot is moving. Feature based localization uses natural environment features as landmarks. This approach has extensively studied for navigation in indoor environment. Although Hough Transform is an effective method for detecting lines and curves from noisy data it has the drawback of sensitive to the resolution of the discretization resolution [6]. A Laser Range Finder is often employed to build a local map which is matched to a global reference map in order to estimate the actual pose of the robot [7, 8].

However most of the proposed solutions do not take into account some real constraints encountered in practical applications. This paper describes an algorithm by which a robot can construct a map on the fly and localize itself in the context of Middle Size League RoboCup considering that the robot motion planning has to require no human intervention during the robot soccer game. Robust localization is the problem of determining the pose of a mobile robot with respect to a global or local frame of reference in the presence of sensor noise, uncertainties and potential failures. A robust and accurate determination of location is fundamental for the motion control of mobile robots. In this paper a robust feature extraction and correspondence algorithm was used to extract the rich features i.e. landmarks from the environment and the landmarks triangulated the robot's positions. The approach constantly updates the robot's pose relative to a global frame i.e. the robot soccer field. The purpose of the landmark detector is to locate candidate landmarks in a range image. These candidates are later provided to the tracker for the purposes of building a set of tracked landmarks.This approach has extensively studied for navigation in indoor environment [7, 8, 9, 10]. However most of the proposed solutions do not take into account some real constraints encountered in practical applications. Self localization methods for mobile robots need to take various sources of uncertainty into account in order to get robust performance.

In this paper, we presented a robust feature extraction algorithm and the matched feature acquired by a Laser Range

* This work is partially supported by NNSF of China Grant #50128504 (Young Oversea Investigator Award) and #50390063 to Qixin Cao and Jay Lee.

0-7803-8273-0/04/$20.00 ©2004 IEEE

Finder is used to estimate the robot pose. The main challenge in this work was on choosing feature representation for many types of geometrical feature. With such representation it is possible to solve pose estimation by means of feature transform. Because map matching is essential for pose estimation, a feature extraction and correspondence algorithm is discussed in a split and merge framework. In this phase we attempt to find a list of local and global features. The approach to find feature correspondences is implemented by a sliding window with a given width. We further addressed the issue of pose estimation employing the list of matched feature. On-line localization is performed by matching candidate landmarks from the robot's current view to the tracked landmarks, and interpolating a parameterisation of the set of tracked candidates. Further, experimental validation was carried out in a real environment with the mobile robot Jiaolong.

This rest of the paper is organized as follows: Section II gives a detail description of the hardware system and sensor configurations. Section III presents the feature extraction and correspondence algorithm, geometrical feature mapping algorithm and self localization algorithm. The experiment results are described in Section IV. Section V gives a discussion and concludes this paper.

II. Mobile Robot and Sensor Configuration

For the Middle-Size League RoboCup, we have developed autonomous mobile robot Jiaolong and a laser range finder is mounted on top of the robot as shown in Fig. 1. The Laser Range Finder used for local map building is a LMS 200 manufactured by SICK Optic Company. In this experiment its maximum range is set to be 8000mm, with a statistical error +/- 15mm. This sensor provides scans in 180° angular field with a resolution of 1° through its RS232 interface with 19200 baud and the frequency is about 5Hz.

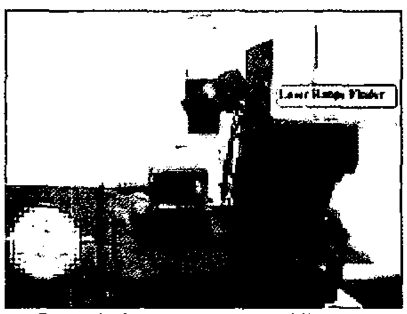


Fig. 1 Jiaolong autonomous mobile robot

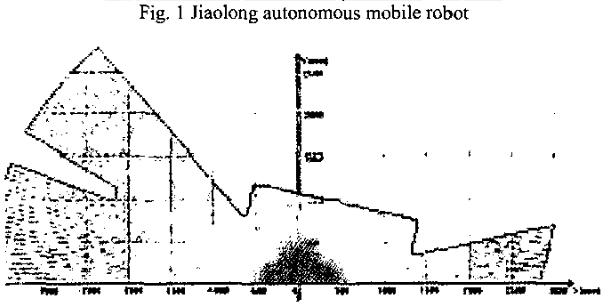


Fig. 2 Laser Range Finder data

III. Algorithms

A geometrical feature is a special type of target that can be reliably observed in successive sensor measurements and that can be accurately described in terms of a concise geometric parameterization[11]. Feature based localization uses natural environment features as landmarks. In order to estimate the pose of the robot, using Laser Range Finder to build a local map has been extensively studied for navigation in indoor environment [7, 8]. However most of the proposed solutions do not take into account some real constraints encountered in practical applications. In the field of RoboCup Middle-size league, the wall of the goal, the goal posts and the corner posts can be used as landmarks. Due to the limitation of Laser Range Finder's view angle, the posts are only partial observable in some occasion. The corner posts are easily confused with the people so it is not meaningful and reliable to employ corner posts to localize the robot.

During the robot soccer game the robot motion planning has to require no human intervention. The real robot soccer field is dynamic and adversarial. Assuming that no collisions occur is unacceptable, while the dead reckoning may generate unbounded error due to the wheel slippage and collision. Landmarks are easily occluded by other robots in the soccer game. Therefore the robot must consider the collisions and occlusions in order to get real time robust performance.

A. *Feature Detection Using Feature Transform*

First we transform the polar coordinate $O(r_i, \theta_i)$ into Cartesian coordinate system (x_i, y_i) , where $x_i = r_i \cos\theta_i, y_i = r_i \sin\theta_i$ (Fig. 2). Next we calculate the forward deference:

$$\delta x_i^f = x_i - x_{i+\eta}, \ \delta y_i^f = y_i - y_{i+\eta}, \tag{1}$$

and backward deference:

$$\delta x_i^b = x_i - x_{i-\eta}, \ \delta y_i^b = y_i - y_{i-\eta}, \tag{2}$$

on (x_i, y_i). The range image is divided into intervals with step length η which is decided by the sensor uncertainty field. The Feature Transform is defined as follows:

$$\delta x_i' = \delta x_i^f - \delta x_i^b, \delta y_i' = \delta y_i^f - \delta y_i^b. \tag{3}$$

B. *Split-and-Merge Feature Extraction Algorithm*

Once the original data is mapped into feature space, the eigenvalue of the intervals are calculated to discriminate different types of geometrical features, which are then used to registry the features in the whole map. In order to improve the robustness of the feature transform and consider the sensor uncertainty implicitly we use an observing window with k width. The eigenvalue in the sensor feature space is defined as

$$F_i = \sum_{j=i}^{i+k} \delta x_j' \cdot \delta y_j', \tag{4}$$

where k is width of the observe window.

The feature type of the interval is decided by its threshold and the precise position of the feature point is calculated by:

$$E_i = \underset{i \in [j, j+k]}{\arg\max} \, |\, \delta x_i' \cdot \delta y_i' \,| . \qquad (5)$$

In order to unify the feature extraction algorithm into a framework we employ the multi threshold method.

$$C_\theta = \{ i \in I \mid E_i \wedge \delta y_i' \leq 0 \wedge (F_i \geq T_{Concave}) \} \qquad (6)$$

$$E_\theta = \{ i \in I \mid E_i \wedge (F_i \geq T_{Edge}) \vee (F_i \leq -T_{Edge}) \} \qquad (7)$$

$$S_b = \{ i \in I \mid (T_{Concave} \geq F_i \geq T_{Segenment}) \vee (-T_{Segenment} \geq F_i \geq -T_{Concave}) \} \qquad (8)$$

$$S_e = S_b + k \qquad (9)$$

Where C_θ is index of the concave corner, E_θ is index of the edge, S_b and S_e are the beginning and end index of the segments. The above detail geometrical features are essential for the geometrical reasoning.

The splitting phase (See Fig. 3) starts from (r_0, θ_0) taking the whole range image as candidate and split the raw range data according to the step length η. After the Feature Transform and the multi threshold segmentation, the interval's type is decided by the Feature Transform and its eigenvalue. Feature vector is defined as

$$L_i (i = 0, 1, ..., n) = (\theta_{iBegin}, \theta_{iEnd}, F), \qquad (10)$$

where θ_{iBegin} and θ_{iEnd} are the beginning and end index of the feature respectively. $F \in (U, E, C, S)$ represents the different feature types. Where U represents that feature of interval is undecided; E represents that feature of interval is Edge; C represents that feature of interval is Concave; S represents that feature of the interval is Segment.

Algorithm: *Split - and - Merge*()

{

Given LRF range data,start from (r_0, θ_0);

divide the image into $[180/\eta]$ intervals ;

perform *Feature Transform* and calculate the *eigenvalue*;

extract feature using *multi threshold segmentation*;

if the interval's feature $F_i = C$ then registry a *concave* corner;

if $F_i = S \wedge F_{i-1} \neq S$ then begin a segment and record θ_{beginS} ;

if $F_i = S \wedge F_{i-1} = S$ then merge the adjacent intervals;

if $F_i = S \wedge F_{i+1} \neq S$ then end a segment and record θ_{endS} .

}

Fig. 3 Split-and-merge algorithm

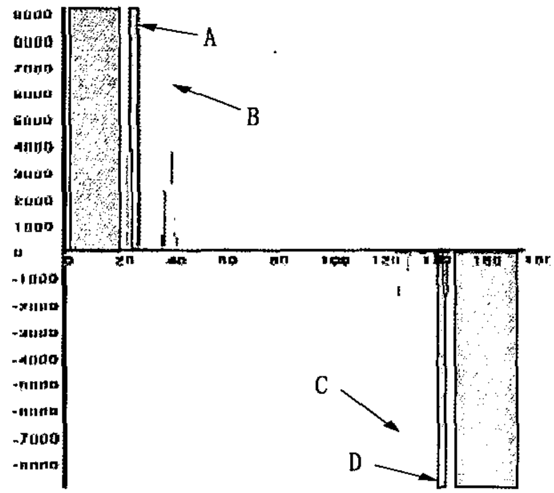


Fig. 4 Eigenvalues of feature extraction

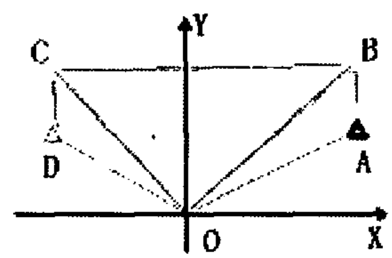


Fig. 5 Extracted features of the environment

After the spilt loop we merge adjacent candidate intervals if they are homogenous and extract task-specific knowledge (corners, straight lines, edges etc). The concave corners are extracted by the multi threshold segmentation and their positions are also decided. At the same time the property of the geometrical feature could be calculated. It merges segments until their eigenvalues are greater than a threshold. The detail algorithm is shown in Fig. 3.

C. Geometric Calculations

Knowing the positions of the two goal posts with respect to the robot, i.e. $A^P(r_A, \theta_A)$ and $B^P(r_B, \theta_B)$ in Fig. 6, and the positions of the goal posts in the field, i.e. (X_A, Y_A) and (X_B, Y_B), it is possible to determine the posture of the robot (X_R, Y_R, θ_R) using simple geometry. θ_R is the angle between $\overline{RY_{LCS}}$ and $\overline{OX_{WCS}}$. However the calculation must concern that the goal posts are not always in the view field of the Laser Range Finder.

In Fig. 6 ρ is the distance from the robot to a line α is the orientation of the line in the Local Coordinate System (LCS) and l is the length of the line.

$$\alpha = tg^{-1} \frac{r_A \cdot \sin\theta_A - r_B \cdot \sin\theta_B}{r_A \cdot \cos\theta_A - r_B \cdot \cos\theta_B} \; (\alpha \in [0, 2\pi]), \qquad (11)$$

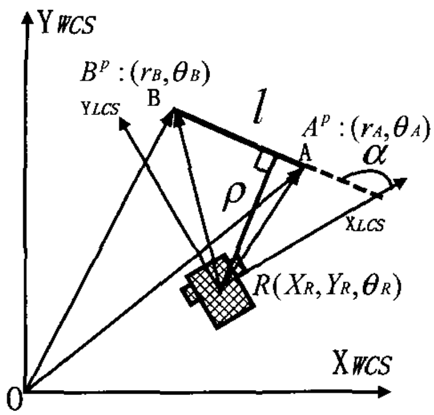


Fig. 6 Triangle Self Localization

$$l=\sqrt{(r_A\cdot\sin\theta_A-r_B\cdot\sin\theta_B)^2+(r_A\cdot\cos\theta_A-r_B\cdot\cos\theta_B)^2}\ ,\tag{12}$$

$$\rho=\frac{r_A\cdot r_B\cdot\sin(\theta_B-\theta_A)}{l},\tag{13}$$

$$\phi=\theta_B-\theta_A.\tag{14}$$

The line model in Local Coordinate System is represented by $AB:\rho-|r_A\sin(\alpha-\theta_A)|=0$. (15)

The extracted features are A, B, C, D, AB, BC, and CD as shown in Fig. 4 and Fig. 5. BC is the principal part of the goal which is the most important candidate landmark for the self localization of the robot.

Step 0: Initializing the parameters such as the thresholds, maximum range of LRF, the Step Length η etc.
Step 1: Request the data from LRF and update the environment;
Step 2: Extract the goal from the background and get the geometrical information
$\delta_r=r_i-r_{i-1}$, where $i\in I$
$O_{end}=\{i\in I\,|\,r_{i+1}-r_i>T_{end}\}$,
$O_{begin}=\{1+i\in I\,|\,r_{i+1}-r_i<-T_{begin}\}$
O_{begin} and O_{end} are the beginning and end post of the goal respectively;
Step 3: Feature Extraction and Correspondence Main Loop;
Step 4: Landmark selection;
Step 5: Geometrical calculation;
Step 6: Record the history list of the pose.

Fig. 7 Self localization algorithm

The candidate principal wall is decided by (l,ρ,ϕ):

$$G=\{i\in M\,|\,\underset{i\in[0,n]}{\arg\min}\,\rho\wedge l\le l_{\max}\wedge l\ge l_{\min}\wedge\phi>\phi_{cr}\}\ .\tag{16}$$

where $l_{\max}=2100, l_{\min}=550, \phi_{cr}=30$.

That is to say the most likely principal wall of the goal is the nearest line with appropriate length and view angle.

Next the robot pose (X_R,Y_R,θ_R) as shown in Fig. 6 is calculated as follows:

$$\theta_R=\alpha+\sin^{-1}((Y_B-Y_A)/\|AB\|),\tag{17}$$

where α is given by (11).

$$X_R=\frac{X_A+r_A\cos(\theta_A+\theta_R)+X_B+r_B\cos(\theta_B+\theta_R)}{2},\tag{18}$$

$$Y_R=\frac{Y_A+r_A\sin(\theta_A+\theta_R)+Y_B+r_B\sin(\theta_B+\theta_R)}{2}.\tag{19}$$

IV. Experiment Setup and Result

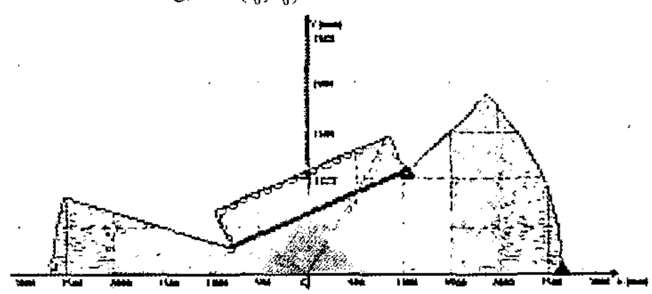
We carried several experimental runs in order to show the robust performance of the localization method.

A. Experiment Setup

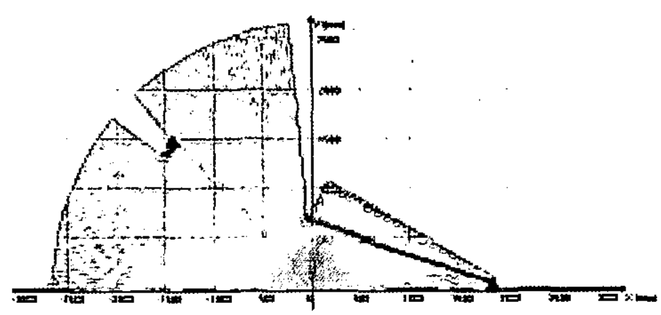
The feature is determined by the eigenvalue after multi threshold segmentation. The thresholds are decided by the relative sensor uncertainty field of the environment. In the following experiments we select the following parameters: $T_{Concave}=1.2\times10^3k$, $T_{Segement}=500k$, $T_{Edge}=2\times10^4k$. In this paper we let k equal to the step length η.

B. Experiment Results

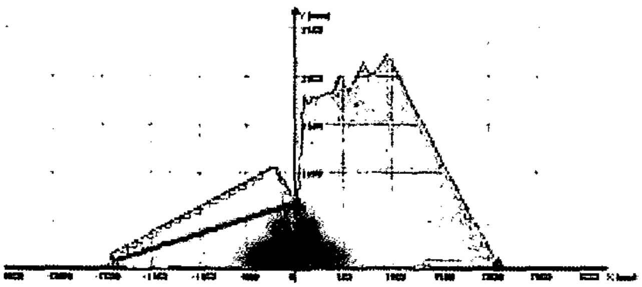
Fig. 8 show the landmark detection result considering different cases and give the robust localization performance. The bottom right triangle marks the beginning of Laser Range Finder's scanning, i.e. (r_0,θ_0).



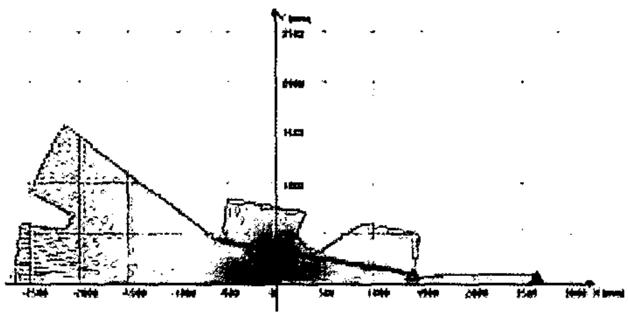
(a) Robot in normal position



(b) The heading of the robot is positively too large and the goal is only partial observable.



(c) The goal is only partial observable and there is a confused wall in the field of the robot.



(d) The goal is occluded by an object.

Fig. 8 Landmark Detection Experiment Result

The history list of the self localization result is shown in Fig. 9, which demonstrates the static and the dynamic localization performance. When the robot stop at a give position the static error is very small as shown in the period of $t \in [70,130]$. The dynamic response is quickly enough when the robot change the pose in the period of $t \in [280,320]$. The allowable heading changing range is $\theta \in [-30,30]$ for the robot.

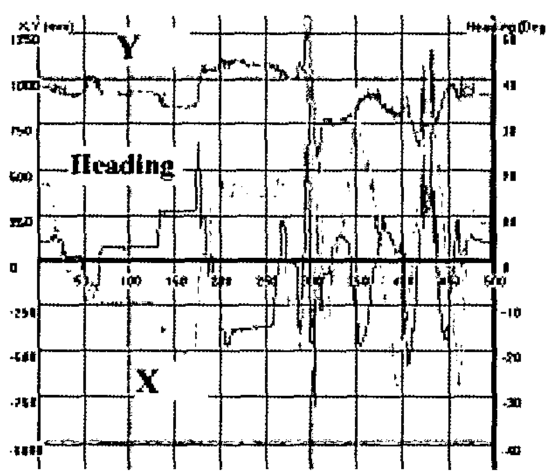


Fig. 9 Self Localization Result

V. Conclusion

In this paper we present a simple but robust feature extraction algorithm using the multi threshold segmentation based split-and-merge method. The performance and the characteristics are studied by the experiment result. Further Feature Transform and feature eigenspace are defined in order to extract the geometrical features and represent the features in a unified framework. The feature extraction and correspondence is accomplished by learning a set of geometrical features called landmarks (i.e. concave corner and straight line, etc.), each of which is detected as a local extremum of a measure of uniqueness and represented by an eigenvalue. After the original data is mapped into feature space, the eigenvalue of the intervals are calculated to discriminate different types of geometrical features, which are then used to registry the features in the whole map.

Then we present an approach to feature-based mobile robot global localization, i.e. the task of obtaining a precise position estimate for a robot, even without an a priori estimate. Our approach combines the strengths of statistical and feature-based methods. Several experimental results on autonomous navigation of the robot soccer in unknown environments show the robustness of the feature transform algorithm and the self localization algorithm in the context of Middle Size League RoboCup.

Acknowledgment

The first author is grateful to Zhen Luo and Shouhong Miao for their valuable help.

References

[1] J. Borenstein, B. Everett, and L. Feng, "Navigation Mobile Robots: Systems and Techniques," Wellesley, MA: A. K. Peters, 1996.

[2] Dieter Fox, Wolfram Burgard, Sebastian Thrun. Active Markov localization for mobile robots Robotics and Autonomous Systems, vol. 25, no. 3-4, pp. 195-207, 1998.

[3] O.Wijk, H.I. Christensen, "Localization and navigation of a mobile robot using natural point landmarks extracted from sonar data," Robotics and Autonomous Systems, vol. 31, no.1-2, pp. 31-42, 2000.

[4] Michael Beetz, Wolfram Burgard, Dieter Fox, and Armin B Cremers. "Integrating active localization into high-level robot control systems," Robotics and Autonomous Systems, vol. 23, no. 4, pp. 205-220,1998.

[5] Axel Großmann and Riccardo Poli, "Robust mobile robot localization from sparse and noisy proximity readings using Hough transform and probability grids," Robotics and Autonomous, vol. 37, no. 1, pp. 1-18, 2001.

[6] Johan Forsberg, Ulf Larsson, Åke Wernersson "Mobile robot navigation using the range-weighted Hough transform," IEEE Robotics and Automation Magazine, vol. 2, pp.18-26, March 1995.

[7] Kai O.Arras, Nicola Tomatis, Björn T.Jensen, and Roland Siegwart. "Multisensor on-the-fly localization: Precision and reliability for applications," Robotics and Autonomous Systems, vol. 34, no. 2-3, pp. 131-143, 2001.

[8] U. Larsson, J. Forsberg, and A. Wernersson, "Mobile robot localization: integrating measurements from a time-of-flight laser", IEEE Transactions on Industrial Electronics, vol. 43, no. 3, pp 422 – 431, 1996.

[9] Nikos Vlassis, Yoichi Motomura and Ben Kröse, "Supervised linear feature extraction for mobile robot localization," Proc. IEEE Int. Conf. on Robotics and Automation, pp. 2979-2984, Apr. 2000.

[10] Clark F. Olson, "A general method for geometric feature matching and model extraction," International Journal of Computer Vision, vol. 45, no. 1, pp. 39-54, 2001.

[11] J. Leonard and H. Durrant-Whyte, "Mobile robot localization by tracking geometric beacons," IEEE Transactions on Robotics and Automation, vol. 7, no. 3, pp. 376–382, 1991.