Relative position estimation in a group of robots

Mark Moors[*], Frank Schneider[+], and Dennis Wildermuth[+]
* University of Bonn, Department of Computer Science III, Bonn, Germany
+ Research Establishment for Applied Sciences (FGAN), Wachtberg, Germany

ABSTRACT

This work presents a new approach to the problem of relative position estimation for multi robot systems. It also creates the basis for establishing and maintaining a common co-ordinate system for a group of robots. The information of a laser scanner system mounted on each of the robots is used to calculate the relative position of each of the surrounding robots. The measured movement of each robot is compared to the reported movement and laser readings, which are communicated between the robots. From this comparison, a co-ordinate transformation is calculated. Once the transformation is calculated, a Kalman Filter is used to track the robots positions. The algorithm is implemented on a real multi-robot system. Preliminary results of real world experiments are presented.

1 INTRODUCTION

When using a multi-robot system in which the robots have to fulfil a cooperative task, two typical positioning problems arise:

- Robust position estimation for each robot itself, and
- Relative position estimation of the other group members.

The first problem is typically solved by some kind of simultaneous localisation and map building (SLAM) method. An important precondition for these methods is that the environment provides enough features (e.g. landmarks) to give the localisation "a handle to grip on". This is not necessarily the case in open space like large hallways or long monotone corridors.
The second problem especially arises when the robots do not have a common co-ordinate system, which is often the case if GPS is not available. Common reference points like landmarks or predefined co-ordinate systems must often be specified by an operator.
Hence, the multi-robot positioning problem asks if it is possible for an autonomous robot to start at an unknown location in an unknown environment, and then to incrementally estimate its own position and the relative locations of the other robots using only sensor information.

The answer would be a robust, fast, and precise method that does not need any preconditions or special assumptions about the environment. This paper presents such an approach to relative position estimation in a group of robots, which is based on sensor and odometry information only. The method is divided into two stages. First, each robot scans its surrounding environment for moving robots. Whenever a robot moves and therefore its position inside the sensor field changes, the movement with respect to the sensing robot's co-ordinate system can be measured. The results of these observations are then communicated to the other robots. By comparing that movement with the one the moving robot itself reports, it is possible to calculate the transformation matrix between the co-ordinate system of that robot and the common relative co-ordinate system.
Based on this information each robot can estimate the relative positions of all visible robots and use them as landmarks to improve its own position estimation. This again results in better estimates for the localisation of the other robots, which in turn results again in a better localisation for the robot itself.
While most localisation techniques are based on global strategies that make use of special landmarks or other kinds of a priori knowledge, the described method uses only local information. It also allows introducing a method for establishing a common co-ordinate system with reference only to the robots themselves. (We will call this a 'relative' common co-ordinate system throughout the rest of this manuscript.) Since it is not possible to map such a relative co-ordinate system to any system of global world co-ordinates, it is of course not useful for all multi robot applications. Nevertheless, for most navigation problems it is sufficient, for example moving in formation and exploration.

2 RELATED WORK

The problem of single robot localisation is widely studied in the literature [12, 13, 14, 15]. Most of the approaches to SLAM can roughly be classified by the kind of sensor data processed and the matching algorithms that are used. One method is to extract landmarks out of the data and match these landmarks to localize the robot in the map being learned. The other set of approaches use raw sensor data and perform a dense matching of the scans. All these approaches have the ability to cope with a certain amount of noise in the sensor data, but it is assumed that the environment is almost static during the mapping process.
In the recent years, the problem was extended to multi-robot localisation [18, 19, 20]. The major difference to these approaches is that most of them use maps and/or landmarks.
Some authors worked on similar concepts using vision in order to reduce the odometry error of a single robot system. Murray [6] and Braithwaite [1] use a movable stereo camera system to follow remarkable points in the surrounding environment. Using the measured distances and rotation angles of the camera, they calculate the current movement of the robot.
In several approaches, these results are transferred from a single to a multi robot system. Some authors add additional global information sources like GPS to achieve greater accuracy [2, 9], whereas others restrict to the robot group itself. Kurazume et al., for example, develop a so-called Cooperative Positioning System (CPS) [4, 5], other similar ideas can be found in [7, 8]. Since in these latter works the aim is to generate and maintain a global co-ordinate system, a great accuracy is needed. Just one robot moves at any given period of time, while the others are standing still, thereby functioning as 'temporary' landmarks. Suzuki and Yamashita [11] present an approach to building a common co-ordinate system in which all robots may move simultaneously, but they use a simulated and somewhat idealised robot system. In their simulation, for example, every robot has a full 360-degree view and is capable of error-free measurement of the relative positions of the other robots.

A quite similar approach to ours can be found in [3]. The main difference is that we use a Kalman-Filter based approach. The particle filter needs in contrast to the Kalman filter much more computing power and is mathematically more difficult to handle.

3 THE ALGORITHM

As described in the introduction a two-step approach is used to estimate the positions and to establish the common co-ordinate system. The first step consists of detecting the moving robots. This information is shared with the other robots and used to estimate their current locations, as well as to improve the own position estimation. In the second step these position information in combination with the robots' movement data is used to establish and share the so-called 'relative' common co-ordinate system.

3.1 Estimating the robots relative positions

The first step of our algorithm is an estimation of the observed robots distance and angle relative to the observer. The observers use SICK-PLS Laser scanners, which generate a 180-degree scan of the area in front of the observer in a frequency of about five scans per second.
Since sensor interpretation was not the main goal of our paper, we used a rather simple method for finding the observed robot in the data, which produced good results as long as the observer himself is not moving. This method will be replaced in the future with a tracking method for moving observers. Based on a history table of the last few scans, which serve as some kind of local map, a moving object in the sensors field can be found quite easily, since the readings in the direction of the moving object have to be shorter than in the local map.

3.1.1 Fixed Observer

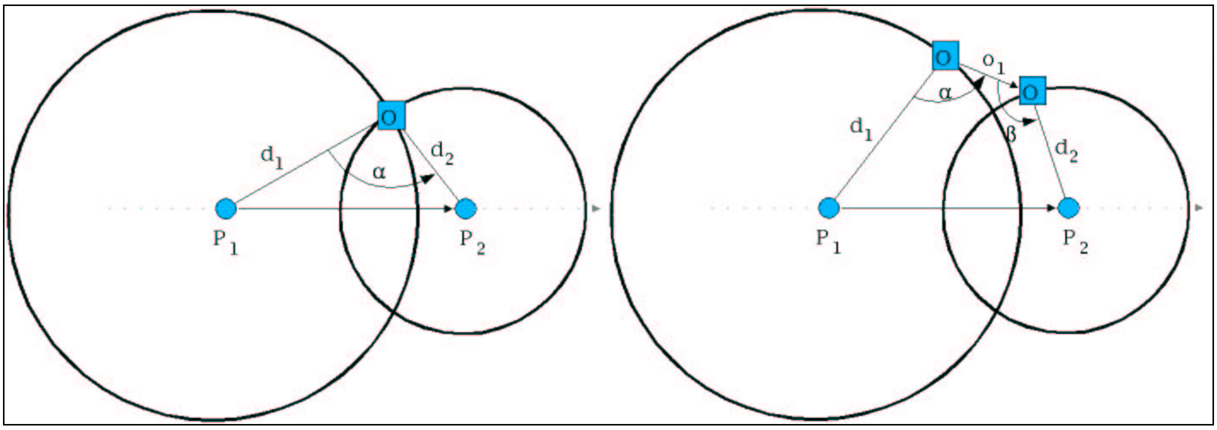

Figure 1 – Decomposition of observer position

Consider a moving robot travelling from point P_1 to point P_2, while being watched from an observer O (Figure 1 left). Given both distances d_1 and d_2, there are two possible solutions for the position of the observer (both intersections of the two circles). Given the leading sign of the angle α one possible solution remains. (Since SICK-Laser scanners number their beams from right to left, calculating the leading sign of α is done by simply subtracting the number of the beam which hit P_2 from the number of the beam which hit P_1.)
Given both positions P_1 and P_2, together with the calculated position of the observer, a transformation matrix A can be calculated, which transforms the coordination system of the observer to the coordination system of the travelling robot. When one robot is travelling, while being watched from multiple overseers, we can calculate such a matrix for each observer, leading to a common coordination system.

3.1.2 Moving Observer

Consider a situation similar to the last paragraph, but let the observer move from a position O_1 to O_2, while the observed robot travels from point P_1 to P_2 (Figure 1 right). Contrary to the situation before, even if we have information about the travelled distances and the measured distances between the observing and the observed robot, there are still infinite possibilities for the relative position of the robots. However, given the angle α and β just one solution remains. As before, we can calculate a transformation matrix A for the observer, which transforms coordinates in the observers coordination system to the coordination system of the observed robot. After calculating this transformation matrix, an observer can transform his estimates of the observed robots position from his personal coordination system to this common coordinate system, which all robots share.

3.2 Using the Kalman filter to track the robot

Once the transformation matrix for an observer is calculated, the observer can make estimates of the tracked robot and transform them to the coordination system of the tracked robot. This estimates can be joined by a Kalman filter leading to a common track-estimate of all tracking observers. We therefore define the state of the tracked robot as vector

$$x(k) = [p_x(k), p_y(k)]^T.$$

The information gained by the odometry is used for the prediction-step of the Kalman filter. So let the control input for the prediction step be:

$$u(k) = [\Delta p_x(k), \Delta p_y(k)]^T$$

(This information cannot be taken directly from the odometry, since it would suffer from errors in the robots angle. The robots real angle can be derived from a small history of positions. Given this correction u(k) can be correctly calculated. This is a somewhat sloppy method, but proves quite okay in practice.)

Then the state prediction can be written as:

$$x^- (k) = [x(k-1)+u(k))]^T$$

The correction step of the Kalman filter is based on the estimation of an observer let the estimation be:

$$z(k+1) = [est_x(k) , est_y (k)]^T$$

So, the correction step can be written as:

$$x(k) = x^-(k) + K (z(k) – x^-(k))$$

whereas K denotes the Kalman gain. For details about the Kalman filter, refer to [16].

4 EXPERIMENTAL SETUP

4.1 Real world experiments

The experiments were conducted in the "Experimental Human Multi Robot System" laboratory of the FGAN. The experimental set up consists of a more or less empty hall of the size 18 x 15 meters, one B21 from iRobot, and four Pioneer I robots from ActivMedia. All robots are equipped with odometry, SICK laser scanner and radio Ethernet. In these first experiments, the four Pioneer robots were used as the observing robots. The B21 was used as the moving target to be observed by the other robots. Four different routes were used in the experiments (Figure 2).

4.2 Simulations

While our main target is to demonstrate our algorithm during real world experiments, we also conducted some simulation runs of the software. Simulation gives us the advantage of a grand-truth trajectory of the observed robots, allowing us to make some statements about the deviation of the estimation compared to the real position of the observed robot. During our simulation runs, we created a somewhat idealistic version of the real experimental hall. It is bigger (20x20 Meters) and quadratic in shape. We used about the same control input and therefore created very similar trajectories. While comparing simulated odometry (with simulated errors) to our observed track is of limited utility, it is interesting to compare the real position in the simulation to our observed track.

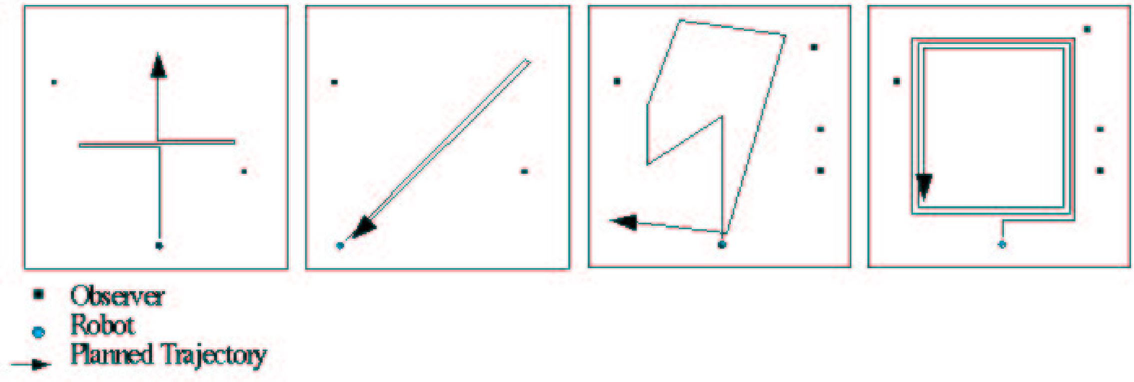

Figure 2 Planned Trajectories

5 RESULTS

5.1 Real World Experiments

As can be seen in figures 3 and 4 our algorithm determines the position of the observers quite correct and tracks the robot over the full distance. Due to the nature of the Laser-Range-Scanner, there is always a small error in the determination of the observed robots position, resulting from the fact, that it is quite difficult to determine the exact angle of the observed robot in the observed coordinate system. This leads to some kind of jitter-effect in the estimations, which could be coped by applying another linear filter.

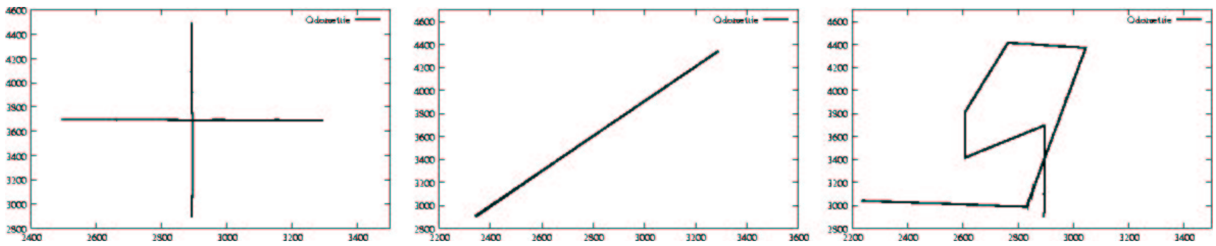

Figure 3 Real world experiments - Odometry Data

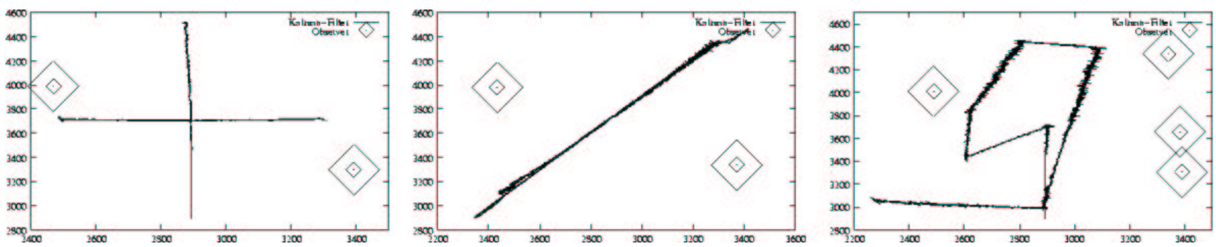

Figure 4 real world experiments - Tracking results

One of the most challenging experiments concerning the odometry is the so-called square experiment. In this set-up, the robot moves on a route that describes a square. Having perfect odometry and traction the robot should exactly arrive at the starting point after completing the square route. Since most of the errors of the odometry result from rotation rather then from translation, in reality it is very unlikely that the robot will close the square. The localisation problem arises from the fact that the odometry does not recognise these errors. In the presented experiments, a typical SLAM method was used for comparison. Figure 5 shows three different outputs of the same track. The little (right) part of figure 5 shows the odometry data, while the bigger (left) part shows track estimates from the Kalman filter (painted in dark grey lines, also recognize the jitter) compared to the estimates of a SLAM Algorithm (painted in light grey dots). The four diamonds represent the estimated observer positions. As can be seen the trajectory of the odometry shows almost no deviation from the original controller input. All tracks are close together. The SLAM algorithm instead gives a quite more realistic view of the real trajectory.

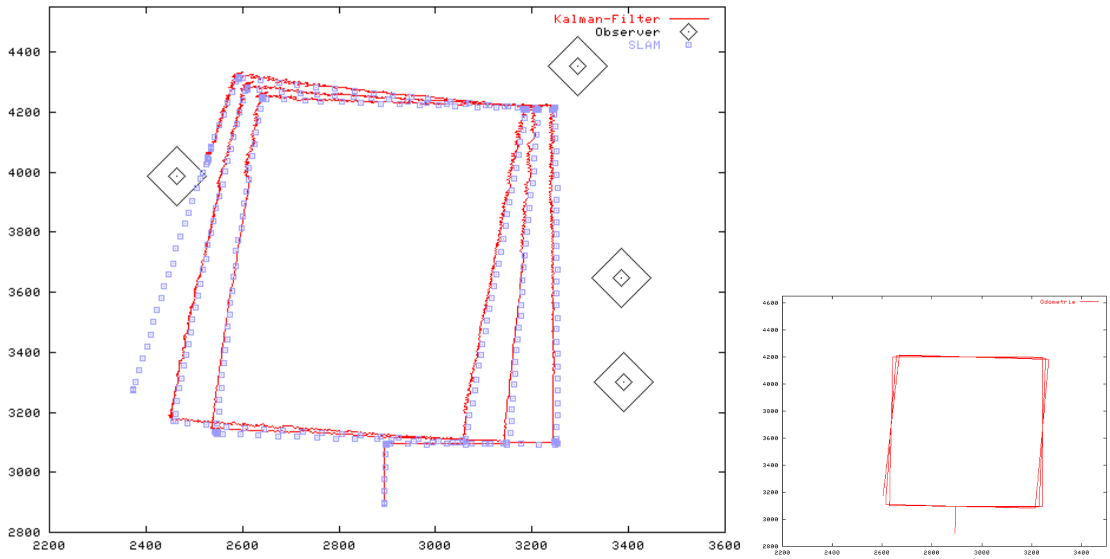

Figure 5 - Square Experiment

The Kalman Filter gives a very good estimate of the observed robots motion. While dead reckoning fails apparently after one full rectangle, our Kalman Filter is able to track the robot for the full distance. The trajectories of the Kalman Filter and SLAM agree on the same trajectory, varying only about 10 cm. Considering the robot's radius, which is about 23 cm, this error is quite small.

The "global" approach of the SLAM will improve as more features are in the area, our approach will improve, as more observers are available. While it is easy to co-ordinate the robots to visit or to stay in each others field of view, providing landmarks is somewhat artificial depending on the environment or application and sometimes even beyond the influence of the operator. The compared SLAM Algorithm is an implementation of laser-scan matching [17].

5.2 Simulation Experiments

As can be seen in figure 6 the filter shows about the same behaviour as in the real world experiments, including the jitter, which results from the errors in the tracking process. Different from the real world experiments, we can draw some quantitative conclusions from this data, since we can use the real trajectory drawn from the simulator itself compared to the estimations of the Kalman-Filter. Calculated over the full tracking time, the filter is never more than 80cm off the tracked target, while the average deviance is less than 11cm.

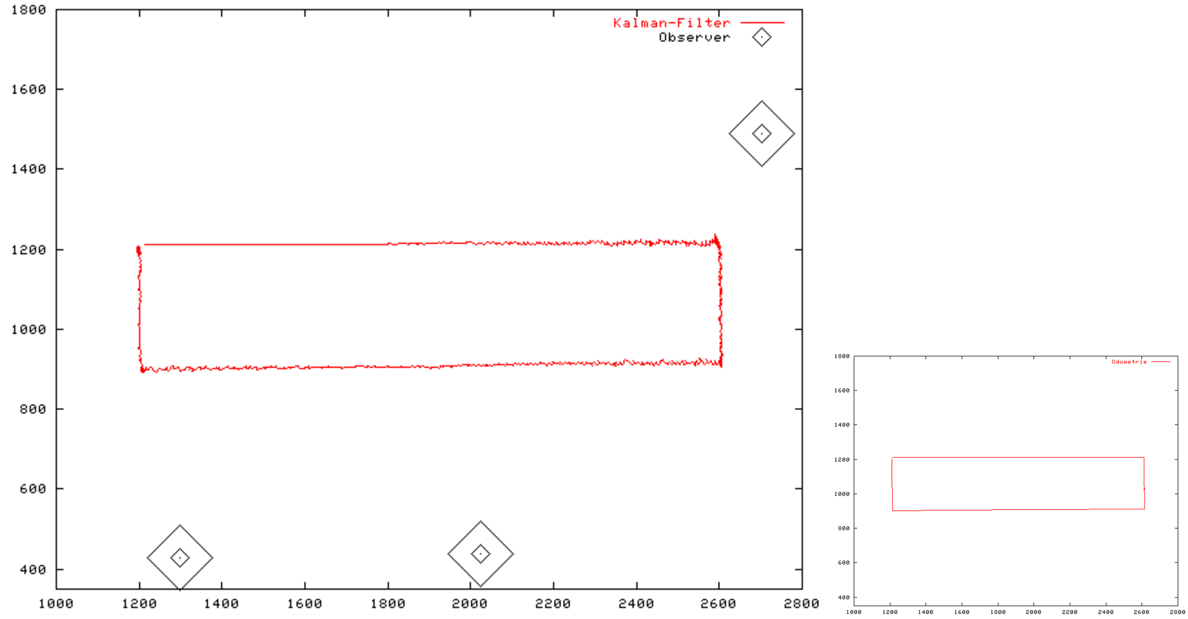

Figure 6 - Simulation Run

6 CONCLUSIONS

Within this work, the authors present a new approach to the problem of establishing and maintaining a common co-ordinate system for a group of robots. A laser range finder in combination with a tracking process is used to calculate the relative position of each surrounding robot. The moving robot uses it as input to a filtering process, which corrects its odometry based position information. This process is implemented by a Kalman filter. In addition to the correction of the moving robot's odometry, also the position and orientation of each observing robot is updated. This position information is used to generate and maintain a so-called relative common co-ordinate system among all robots in the group.
First experiments with real robots were performed and produced promising results. The tracking process generates consistent position information for the other visible robots. It is shown that the resulting position information for the moving robot is comparable in accuracy to other well-known localisation strategies.

7 REFERENCES

[1] R. N. Braithwaite and B. Bhanu. *Robust Guidance of a Conventionally Steered Vehicle Using Destination Bearing*. Proceedings of the 1993 International Symposium on Intelligent Control, Chicago, USA, September 1993.

[2] P. Goel, S. I. Roumeliotis and G. S. Sukhatme. *Robot Localization Using Relative and Absolute Position Estimates*. Proceedings of the 1999 International Conference on Intelligent Robots and Systems (IROS), Kyongju, Korea, 1999.

[3] Andrew Howard, Maja J Mataric´ and Gaurav S. Sukhatme, *Localization for Mobile Robot Teams Using Maximum Likelihood Estimation*, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, EPFL Switzerland 2002

[4] R. Kurazume and S. Nagata. *Cooperative Positioning with Multiple Robots*. Proceedings of the 11th IEEE International Conference on Robotics and Automation. San Diego, USA, May 1994.

[5] R. Kurazume and S. Hirose. *An Experimental Study of a Cooperative Positioning System*. in Autonomous Robots 8 (1), pp. 43-52, Kluwer Academic Publishers, January 2000

[6] D. W. Murray, I. D. Reid and A. J. Davison. *Steering and Navigation Behaviours using Fixation*. British Machine Vision Conference, 1996.

[7] I. M. Rekleitis, G. Dudek and E. E. Milios. *Multi-Robot Exploration of an Unknown Environment, Efficiently Reducing the Odometry Error*. Proceedings of the International Joint Conference in Artificial Intelligence. Nagoya, Japan, August 1997.

[8] I. M. Rekleitis, G. Dudek and E. E. Milios. *Reducing odometry error through cooperating robots during the exploration of an unknown world*. Proceedings of the 5th IASTED International Conference on Robotics and Manufacturing. Cancun, Mexico, May 1997.

[9] S. I. Roumeliotis and G. A. Bekey. *Collective Localization: A distributed Kalman filter approach to localization of groups of mobile robots*. Proceedings of the 2000 IEEE International Conference on Robotics and Automation (ICRA), San Francisco, USA, 2000.

[10] D. Schulz, W. Burgard, D. Fox, and A. B Cremers. *Tracking multiple moving targets with a mobile robot using particle filters and statistical data association*. Proceedings of the 2001 IEEE International Conference on Robotics and Automation (ICRA), 2001.

[11] I. Suzuki and M. Yamashita. *Agreement an a Common X-Y Coordinate System by a Group of Mobile Robots*. Proceedings of the Dagstuhl Seminar on Modeling and Planing for Sensor-Based Intelligent Robot Systems. Dagstuhl, Germany, September 1996.

[12] G. Dissanayake, H. Durrant-Whyte, and T. Bailey. *A computationally efficient solution to the simultaneous localization and map building (SLAM) problem*. In ICRA'2000 Workshop on Mobile Robot Navigation and Mapping, 2000.

[13] J.-S. Gutmann and K. Konolige. *Incremental mapping of large cyclic environments*. In Proc. of the IEEE Int. Symp. on Computational Intelligence in Robotics and Automation (CIRA), 1999.

[14] F. Lu and E. Milios. *Globally consistent range scan alignment for environment mapping*. Autonomous Robots, 4:333–349, 1997.

[15] S. Thrun. *A probabilistic online mapping algorithm for teams of mobile robots*. International Journal of Robotics Research, 20(5):335–363, 2001.

[16] Kalman, Rudolph, *A New Approach to Linear Filtering and Prediction Problems*, Transactions of the ASME--Journal of Basic Engineering, 1960

[17] .Thrun, S. and Burgard, W. and Fox, D., *A Real-Time Algorithm for Mobile Robot Mapping With Applications to Multi-Robot and {3D} Mapping*, Proceedings of the IEEE International Conference, 2000

[18] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun;. *Collaborative multi-robot exploration.* In Proc. of the IEEE International Conference on Robotics &Automation (ICRA), 2000

[19] D. Fox, W. Burgard, H. Kruppa, and S. Thrun.; *Collaborative multi-robot localization.* In Proc. of the German Conference on Artificial Intelligence(KI) and the 21st Symposium on Pattern Recognition (DAGM), Germany, 1999.

[20] Ioannis M. Rekleitis and Gregory Dudek and Evangelos Milios; *Multi-Robot Cooperative Localization: A Study of Trade-offs Between Efficiency and Accuracy*; IEEE/RSJ/ International Conference on Intelligent Robots and Systems, p. 2690-2695, Switzerland, 2002