Robustness and Precision Analysis in Map-Matching based Mobile Robot Self-Localization

Manuel Gouveia[1], António Paulo Moreira[1,2], Paulo Costa[1,2], Luís Paulo Reis[1,3] and Marcos Ferreira[1]

[1] FEUP – Faculdade de Engenharia da Universidade do Porto, Portugal
[2] INESC-P – Instituto de Engenharia de Sistemas e Computadores do Porto, Portugal
[3] LIACC – Laboratório de Inteligência Artificial e Ciência de Computadores da Universidade do Porto, Portugal
Rua Dr. Roberto Frias, s/n 4200-465 Porto, Portugal
{mgouveia, amoreira, paco, lpreis, ee04128}@fe.up.pt

Abstract. In this paper an accuracy analysis is presented for a localization algorithm based on maps correspondence. Initially the algorithm was developed by Martin Lauer [1], for robotic soccer purposes, and it has now been adapted to work on indoor robots localization, using walls and other known fixed objects as references. Infrared range sensors (IR) - for low cost applications - and Laser Scan Range Finder (LS) - for other applications - were used in our approach and compared experimentally. Precision and robustness analysis was made using increasing amount of laser points (which simulates a varying number of sensors) and results were compared to using similar number of IRs. Several tests were conducted with a real robot along a well defined trajectory, following a line previously painted on the floor. Visual markers were placed along the line allowing to keep track of the robot's real position with high precision. The experimental results achieved showed that the implemented algorithm works well with a variety of sensors. The results also show that the scan laser range finder is by far the best type of sensor to measure the range of objects indoors. Using only few points the algorithm achieved good results and demonstrated a very high robustness rate and precision.

Keywords: Mobile Robot Localization, Localization Algorithms, Sensor Fusion, Global Localization

1 Introduction

The main objective of mobile robotics is to develop autonomous and intelligent systems which can operate in real environments and unknown situations. Among the various lines of research carried out in the area of mobile robotics, the functions relating to mobility, such as localization and navigation, have great importance in the development of autonomous and intelligent systems. The localization of mobile robots has involved intense scientific and technological research, and a wide variety of techniques, based on different physical principles and localization algorithms, have already been developed, implemented and studied.

Mobile robot localization is probably one of the most important problems in robotics. Mobile robots are currently used in several applications, as guides in museums [2], [3], for entertainment purposes [4], for indoor cleaning [5], for monitoring patients in a hospital environment [6], [7], as well as in a range of other applications.

In any of these applications, if the mobile robot does not know where it is, how will it be able to complete the task assigned to it? Indeed, it needs to be able to ascertain its location using complex localization algorithms, which require information about the robot's surrounding environment. This information is obtained from sensors, which must have appropriate characteristics for the environment in which they are operating and the information that is needed, while the accuracy of such information is a key issue in ascertaining location.

Localization in a structured environment is based generally on external elements called landmarks. We can use the environment's natural landmarks, or deliberately placed artificial landmarks. The challenge of localization can be easily solved if artificial landmarks or beacons are used [8], [9] together with expensive sensors [10], such as a scan laser range finder [11], which allow large amounts of data to be obtained and which make high precision measurements. Other researchers use natural landmarks, such as walls to solve this problem [12], [13], others have explored other techniques based on vision systems that enable certain visual elements to be identified from the image obtained [14], [15].

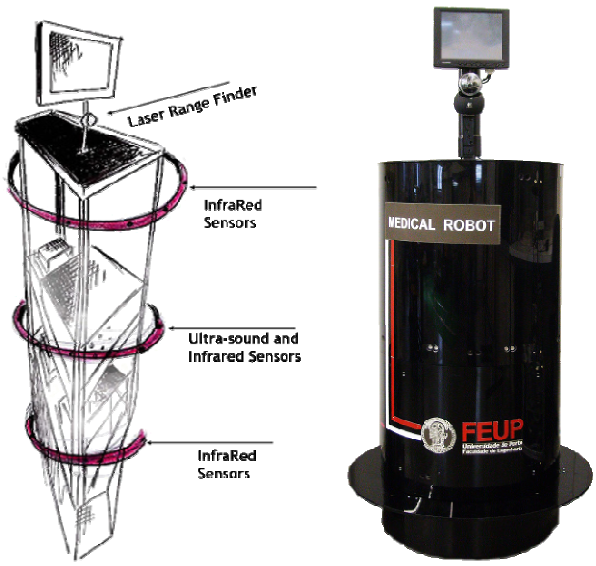

Fig. 1. a) Robot structure and sensor distribution (sketch); b) Medical Robot (foto)

One of the most challenging problems is the difficulty of estimating, with reliability and accuracy, the overall position of the robot in a specific environment.

For example, when the robot is placed in a known environment, it must be able to locate itself without knowing its original position, and also keep track of its local position. Knowing its overall position, the robot can navigate with reliability and, tracking its position locally, the robot can navigate efficiently and perform the tasks assigned to it. To make this possible it is necessary to estimate the status of the robot in real time, based on its initial status and on all the information obtained by its sensors. Therefore, localization is a major obstacle to giving robots real capacity for autonomy.

The aim of this paper is to ascertain whether it is possible to use a method of map matching for robot localization, in real time, using natural landmarks and using low cost sensors. The localization algorithm used is based on [1], it was adapted to be used with other types of sensors than those for which it was originally designed.

The analysis of these algorithms accuracy will be carried out using two kinds of sensors with different range characteristics: Laser range finder and Infra Red. These sensors are not used simultaneously, since the objective is to check their performance independently, especially the Infra Red. The aim of this work is to verify the error and precision in the localization process using these type of sensors.

The experiments were performed in a corridor using the two sensors with data fusion with odometry. Medical Robot (Fig.1) was used [6] as the robot. This robot was built based on the 5dpo soccer robots [16].

This paper is organized as follows: Section 2 presents the main algorithm and the adjustments made to it; Section 3 presents the experimental results achieved; Section 4 presents the work related to localization algorithms with different types of sensors and environments; and Section 5 presents the conclusions of this paper and points out some future work directions.

2 Self-Localization Algorithm

The algorithm implemented was first developed by M. Lauer et al. [1] for the Tribots robot soccer middle-size team [22] to obtain the localization of the robot in a well structured environment, a robotic soccer field, using the field lines. This approach uses an omni-directional camera as the robot's main sensor. The image processing result is a list of positions relative to the robot position and the robot's heading of detected line points.

We have adapted this approach for indoor environments and thus propose some adjustments in the global localization. Instead of the known fields lines we use the walls and other objects in the environment as reference marks represented in a map.

First, the algorithm was implemented and tested in the same conditions verified in [1], on a robotic soccer field, where we verified its reliability and robustness. This also allowed us to adjust the parameters of the algorithm to obtain better results.

For global localization we propose solving the problem in a different way, by estimating the localization for **n** random positions of the robot on the map, for every iteration of the program. For each position the algorithm is applied using the current sensorial information, and **n** localization estimates are calculated, as alternatives to the current localization. Making the comparison of the cost function of each estimate, it is possible to determine which one is better. Because in each iteration, the current robot localization is estimated from its last known position, to obtain global localization and to ensure that it does not jump constantly, the best of all random estimates is saved and compared continuously with all newly obtained random estimates. If that estimate is better than the current estimate, within a given range (5% better, for example), for N consecutive iterations, the robot localization jumps to the new estimated localization. By using repeated random alternative estimates we successively scan the whole map. After several trials and adjustments we concluded that the number of random points, **n**, should be 10, to ensure global localization and obtain appropriate processing time; the new random estimates have to be 10% better for 10 consecutive iterations, to allow the localization to jump to that new estimate. Note that the algorithm implemented in [1], only checks the random positions when the robot start running, and when the localization is assumed to be wrong.

The fusion of data between the estimated localization given by the algorithm and the robot odometry is accomplished by analyzing their uncertainty. Despite the good results we can obtain using only the algorithm, odometry information will always be important, in order to obtain the perception of movement and to guarantee correct localization, especially in the cases of ambiguity which may occur and/or noisy measurements.

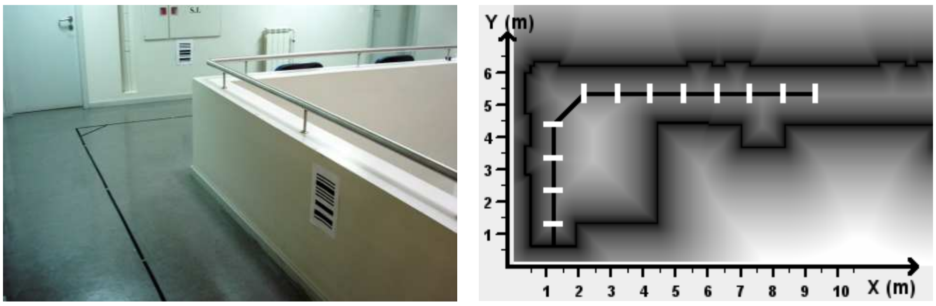

Fig. 2. a) Corridor perspective where the tests were made, with the line on the ground. b) Graphical representation of matrix map used by the algorithm for the corridor and the path and markers used to get real positioning

The map used is also another important issue, because it has the peculiarity of being a matrix, which, in each cell, contains the closest distance to the object on the map. That is, being able to represent each object of the environment with lines and arcs, and considering a cell size for the map, it is possible to gauge the distance of each cell to the nearest object (line, arc) and save its value in a matrix of distances. In Fig. 2b) we can see a graphical representation of the matrix map, in grayscale, where black represents the smallest distance and white is the maximum distance found in the matrix map.

The black line (in the middle of the map) represent the path that the robot follows in every run test we made, and the white marks (rectangles), perpendicular to the black line, are a representation of the markers placed on the ground to give the robot its real localization.

Moreover, we also use other types of sensors able to obtain information used by the algorithm: range of objects present in the environment and represented in the matrix map of distances. Therefore, the main differences of the algorithm presented by [1] are the environment and the type of sensors used to gather information about that environment.

3 Experimental Results

We used two different type sensors: the scan laser range finder [21] and the infrared sensor.

To perform the experiments and get all the information we need, like the robot real position in the environment, we put a line on the ground for the robot to follow it, see Fig 2. The path run 5m in y turns 90º and then runs 7m in x, performing a total of 12m traveled in each run. That line is seen by a camera and with image processing we get its angle and distance relative to the center of robot. With a “FollowLine” control, the robot follows the path we made. Then at every meter we put a marker that can be identified with the image processing, Fig 2. That known marker position, and its orientation relative to the robot body, gives us the exact position and orientation in the moment the robot passes through it. That method of obtaining the robot real position is independent of the algorithm used and was made only to have an external reference to obtain the algorithm error.

Several experiment runs, eleven used for data gathering, were performed and in each one, the data from the odometry, every 11 infrared sensors, 97 laser sensor ranges, were logged at every 40ms, all at the same time and robot real positions was logged when a marker was identified. That will permit a better comparison between the sensors. In every run we used the first mark to define the exact position of the robot.

The program was made in a way that we can run all that logs saved in offline mode (without the robot) as if we were running the robot, remember that we saved all the data needed to do it. That gives us the possibility of repeatability, so that we can run every test how many times we want, and process the data we want to get, like the error of the algorithm.

The matrix used by the localization algorithm, has a resolution of 4 centimeters per cell, although this does not mean that the algorithm give discrete results spaced by 4cm, it is graphically represented in Fig. 2b).

This matrix is pre-calculated and can be stored in a file on disk. This is loaded to memory when the program starts running, along with other required matrices, such as the gradient matrix also used in the algorithm. This allows the algorithm to be fast and efficient, because there is no need to recalculate the matrix of distances for each iteration, the program merely reads the cells of interest in the matrix map from memory, which is an extremely fast process.

The study of experimental data obtained and stored in logs allowed us to compare the error and precision of the algorithm in a real environment for the two sensors used.

3.1 Results using only Odometry

First we run all the data using only odometry, and as we expected the error was high. In global coordinates we get an average error in X of 0.103m, in Y of 0.476m and in orientation of 7,281 degrees. The error grows everytime the robot moves, Fig. 3 and Fig. 5a).

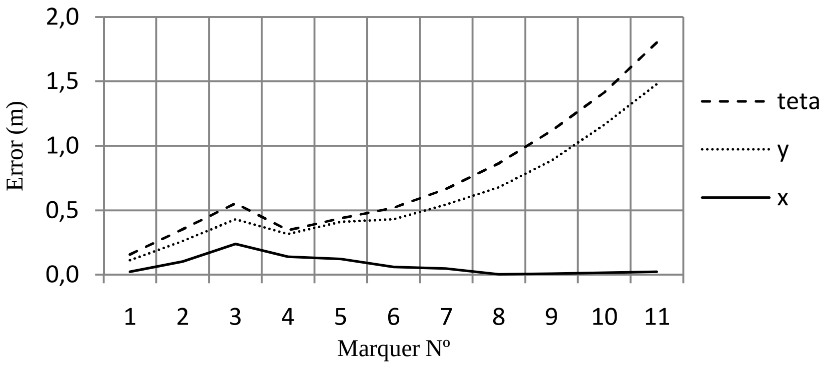

Fig. 3. Evolution of error for a run experiment, odometry only

3.2 Laser Range Finder

The Laser Range Finder, can range in a semicircle of 240º, up to 4m with high accuracy. With this sensor, we can select the number of equally spaced points whose range we want to measure. Accordingly, we measured a maximum of 97, out of 681, equally spaced points in the aperture angle of the sensor. Note that if in one orientation of measurement there is no obstacle it results in a null range. For testing proposes we run the algorithm with 97 points, 49 points, 25 points, 13 points and 7 points in order to examine how the number of points measured impacts the precision and error of the implemented algorithm. We also wanted to compare them directly with the 11 infrared sensors used, although the directions were not the same, as beyond the number of sensors we also want to study the influence of their position, orientation and the spacing between them.

The test runs with 97 points showed that the algorithm can localize the robot even when the robot's odometry is not available for data fusion, even with 49 and 25 points we get the same results, but we want to test it with odometry fusion. Table 1 shows the final results, the average error for the position and orientation given by the algorithm.

Table 1. Average (for all 11 tests, and 11 markers) value of the error for x, y and theta for the laser case, comparing with different number of points

Number of Points	**97**	**49**	**25**	**13**	**7**
Error_x (m)	0,0393	0,0404	0,0419	0,0415	0,0492
Error_y (m)	0,0572	0,0539	0,0454	0,0428	0,0376
Error_theta (degrees)	1,3452	1,2251	1,6023	1,2720	1,4882

Although it was predicted a sharp increase in precision with the number of points used, strangely this does not always happen, which may denote a path, environment (map) and physical localization of sensors, dependency. Note that although in X coordinates the average error always increases with the decrease in the number of points, Fig. 4, this is not the case with the angle and surprisingly the average error in Y coordinates decreases with the number of points. Is expected that the robustness of the algorithm decreases as we use fewer points. I.e. the probability of a total loss of global localization is higher.

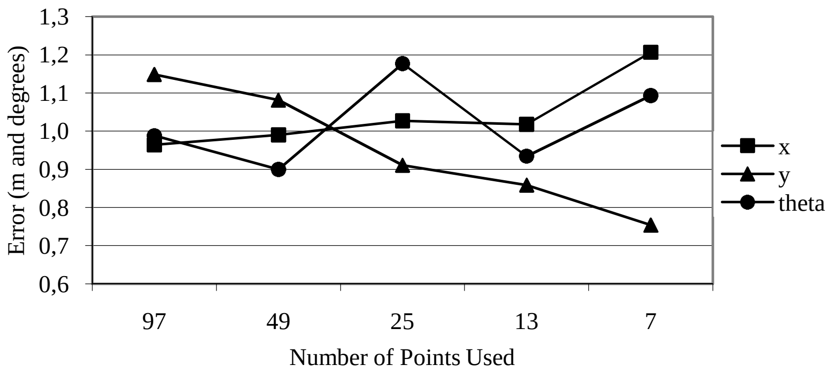

Fig. 4. Error for different number of laser sensor points used

3.3 Infrared range sensors

The Infrared sensor, can range up to 1.5m, with less accuracy and much higher levels of error than the laser range finder. We have 11 sensors mounted around the robot, measuring distances at every 30º. Because of the physical position of these sensors in the robot, Fig 1, the map is different. Fig. 5a).

These 11 sensors are sufficient to properly locate the robot in the sense that we don't have catastrophic failures or errors. Obviously the precision is less than the one obtained with the scan laser. To compare them we interpolate the results from the test runs with 7 and 13 points.

The error obtained from the tests result that in global coordinates we have a average error in X of 0.059m, in Y of 0.089 and in orientation of 4.24º. In one of the markers the robot was in every run a higher error when compared with the other positions, not considering that marker the maximum error in Y drops to half. And so we have a maximum error in X 0.126m, in Y of 0.111m and in theta of 5.58º.

3.4 Comparison

In Fig. 5 we can see an example of a test run (a representation). In each detected mark we can see the robot real position and orientation (at yello) and the localization given by the algorithm evaluated (at red), the yellow dots represent the sensor points position.

As can be seen with odometry only, Fig. 5a), after some meters of movement the robot localization starts to be wrong, when we apply the algorithm with the infrared sensors, Fig. 5b), the localization is much better, maintaining the robot near its real localization. If we look to the Fig. 5c) and 5d) we can see that the error is even less, in the case of the laser range finder. This happens because the sensor accuracy is better than the infrared, and the range of 4m when compared to the 1.5m range of the infrared also influences the localization.

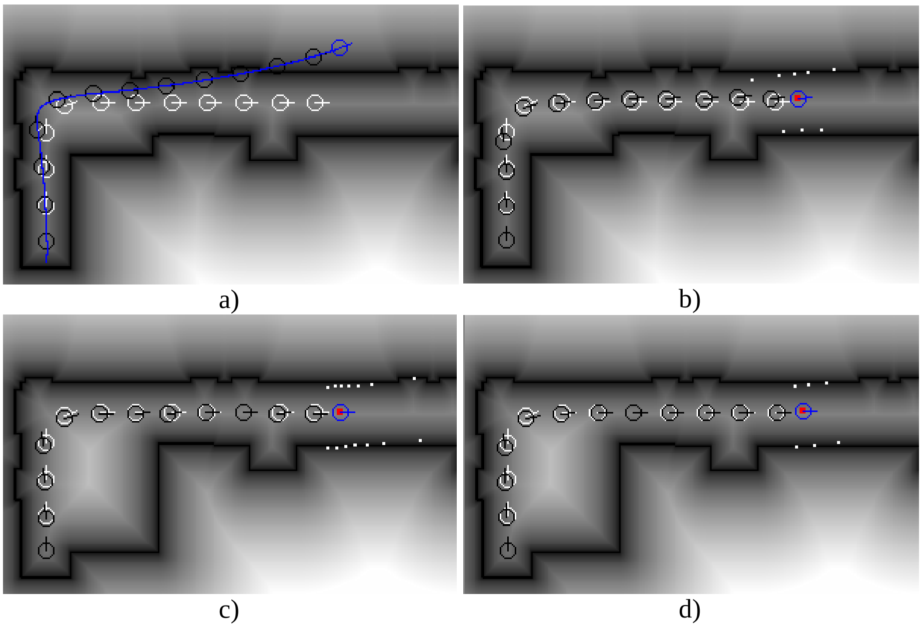

Fig. 5. Representation of path followed by the robot and error of the algorithm: a) with odometry only; b) applying the algorithm with the 11 Infrared sensors; c) applying the algorithm with 13 points of the laser; d) applying the algorithm with 7points of the laser.

Looking at Fig. 5c) and 5d) the difference between them is not noticeable, but comparing with the infrared is. If we look to the values, Table 2, there is a difference of a few millimeters; the error in orientation is more noticeable.

Table 2. Average value of the obtained error of all test runs

Error	X(m)	Y(m)	Theta(degrees)
Odometry	0,1026	0,4762	7,2812
IR	0,0594	0,0887	4,2491
Laser 13	0,0415	0,0428	1,2720
Laser 7	0,0492	0,0376	1,4882

4 Related Work

Several methods have been proposed, ranging from methods based on maps [17], constructed by the robot or pre-programmed, to methods based on recognition of beacons [8] or labels with pre-set positions. The localization problem can easily be solved by placing references in the environment which are distinguishable from all other previously existing objects. However, this solution has its drawbacks, given that the introduction of some changes to the environment may have a negative impact on the environment (for humans). We therefore propose that the environment's own characteristics are used to localize the robot, such as walls and other objects that already exist in the environment.

Typically, these methods have a number of negative characteristics such as being computationally demanding, requiring considerable exploratory time and being inflexible. The localization algorithms [18] that have achieved the best results in robot localization, when compared to other existing algorithms, are those based on the Monte Carlo algorithm [12].

The proposed algorithm in [1] uses a minimization algorithm, Resilient Propagation [19], that is also used in neural networks. This allows rapid convergence of the localization estimate, achieving a more probable location of the robot, with a minimum of error. This method is based on map matching and the major innovation of the method is the fact that the map is pre-calculated, meaning that the values of the distances which are used by the map matching are pre-calculated. These distances are stored in memory in a matrix, which is the map of the algorithm. The gradients of these distances, which are used to minimize the localization error, are also pre-calculated. All of this results in fast processing time and also quickly gives the second derivative of the cost function, allowing us to ascertain the level of uncertainty and the estimated variance of the localization given by the algorithm and therefore merge it correctly with the odometry. As we can read in [1] there are other approaches that are closely related to this one. For example Cox [20] that also uses range finders to detect the walls. But this new approach tackles and improves its bad aspects by giving global localization, a better optimizer, dealing better with the outliers, the robot can move faster, the process time is smaller and any slippage in the odometry is compensated with the algorithm itself.

5 Conclusions

With this research, we verified that the used algorithm works well with a variety of sensors, which are capable of measuring distances to the defined landmarks, whether natural or artificial.

The results achieved are quite satisfactory and enable us to conclude that, for the predicted application, a service robot in a non-industrial environment without the need for high precision, with a minimum of eleven low cost sensors, we have enough robustness and precision, even considering that some of these sensors are giving wrong measurements.

As demonstrated, the algorithm can successfully be used in other applications and with other types of sensors beyond robotic soccer and the video camera sensor, for which it was originally designed.

The results show that the scan laser range finder is by far the best type of sensor to measure the range of objects indoors. Using only few points the algorithm achieved good results and demonstrated a higher robustness rate and precision than the infrared sensor, but at a higher price, which makes it unavailable for most applications.

With this study, we also concluded that the number of points whose range is measured is important; higher numbers of points result in better robustness for global localization, only needing a few points for local localization with odometry. The accuracy and range of measurements is also important because more accurate measured points allows better map matching.

Future work includes adjusting the algorithm to use the maximum number of points possible for global localization but fewer points for adjust of local localization with the fusion with odometry. to verify the accuracy and precision of the algorithm and to evaluate the localization error obtained with it. Also, we intend to tune the weight of the fusion between location and odometry for the IR as the optimal value may be different for the Laser. Other line of future work includes improving the algorithm in the way to better reject the “outliers”, meaning the bad measurements. Finally, we intend to improve the robustness when the robot faces objects that are not covered in the map, like for example moving objects such as people or animals.

Acknowledgments

This research was supported by the Portuguese Science and Technology Foundation (FCT) under the project "ACORD – Adaptative Coordination of Robotic Teams" (PTDC/EIA/70695/2006), funded through the POCI/FEDER program.

References

1. M. Lauer, S. Lange and M. Riedmiller., "Calculating the Perfect Match: an Efficient and Accurate Approach for Robot Self-Localization." *In RoboCup 2005: Robot Soccer World Cup IX , LNCS.* Springer, 2005, pp 142-153

2. S.Thrun, M. Bennewitz, W. Burgard, A. B. Cremers, F. Dellaert, D. Fox, D. Hnel, C. Rosenber, N. Roy, J. Schulte and D. Schulz, “Minerva: a Second-Generation Museum Tour-Guide Robot” *In Proceedings of the 1999 IEEE International Conference in Robotics and Automation*, Detroit, MI, 1999 vol.3, pp.1999-2005
3. D. Fox, S. Thrun, W. Burgard, and F. Dellaert, "Particle Filters for Mobile Robot Localization." *In Sequential Monte Carlo Methods in Practice.* 2000. pp. 470-498.
4. B. Traub A. John D. Schraft and R. D. Graf. “A Mobile Robot Platform for Assistance and Entertainment”. *In Industrial Robot Journal*, 28:83-94, 2001.
5. A. Sousa, P. Moreira and P. Costa, "Multi Hypotheses Navigation for Indoor Cleaning Robots”, *In Proceedings of the 3rd International Workshop on Intelligent Robotics* IRobot 2008 October 14th, 2008, Lisbon University Institute ISCTE, Portugal, pp 71-82.
6. T. Costa., "Medical Robot." Tese de mestrado em Engenharia Electrotécnica e de Computadores, Faculdade de Engenharia da Universidade do Porto, Portugal, 2008.
7. Yulun Wang, Steven E. Butner, and Ara Darzi. “The Developing Market for Medical Robotics” Vol. 94, No. 9, September 2006, pp.1763-1769.
8. J. S. Esteves, A. Carvalho and C. Couto., "Generalized Geometric Triangulation Algorithm for Mobile Robot Absolute Self-Localization." *In IEEE International Symposium on Industrial Electronics (ISIE).* Rio de Janeiro: s.n., 2003. Vol.1, pp. 346-351
9. M. Betke e L., Gurvits., "Mobile Robot Localization using Landmarks." *IEEE Transactions on Robotics and Automation.* 1997. Vol. 13. vol.13, no.2, pp.251-263
10. J. G. Mota and M. I. Ribeiro, “Mobile Robot Localization on Reconstructed 3D Models,” *Journal of Robotics and Autonomous Systems*, 2000. Vol. 31, no. 1, pp. 17-30,
11. Hee Jin Sohn, Byung Kook Kim, “A Robust Localization Algorithm for Mobile Robots with Laser Range Finders”, *In Proceedings of the 2005 IEEE International Conference on Robotics and Automation, ICRA* 2005. 18-22 April 2005, pp. 3545 – 3550.
12. F. Dellaert, W. Burgard and S. Thrun. "Monte Carlo Localisation for Mobile Robots." *In IEEE Int. Conf. on Robotics and Automation (ICRA99).* 1999. Vol.2, pp.1322-1328
13. J. J. Leonard e H. F. Durrant-Whyte, "Directed Sonar Sensing for Mobile Robot Navigation." *Kluwer Academic.* Boston : s.n., 1992.
14. A. Sousa, P. Costa, A. Moreira, A. Carvalho. "Self localization of an Autonomous Robot: Using an EKF to Merge Odometry and Vision Based Landmarks." *Proceedings of the IEEE Int. Conf. on Emerging Technologies and Factory Automation.* 2005. pp. 227-234.
15. D. Cabecinhas, J. Nascimento, J. Ferreira, P. Rosa, P. Lima. "Self-Localization based on Kalman Filter and Monte Carlo Fusion of Odometry and Visual Information." *Robótica.* 2006. No. 65
16. 5dpo Robotic Team Home Page, “5DPO Robotics”. [Online] available at: http://www.fe.up.pt/~robosoc, consulted on June 2009.
17. N. Karlsson, E. Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian and M. Munich. "The vSLAM Algorithm for Robust Localization and Mapping." *In International Conference on Robotics and Automation.* ICRA 2005. pp. 24-29
18. S. Thrun, W. Burgard and D. Fox., "Probabilistic Robotics" *The MIT Press.* 2005.
19. M. Riedmiller and H. Braun, "A Direct Adaptative Method for Faster Backpropagation Learning: the RPROP Algorithm." In Proceedings of the IEEE International Conference on Neural Networks. 1993. pp. 586-591.
20. I. J. Cox, “Blanche – an Experiment in Guidance and Navigation of an Autonomous Robot Vehicle”, IEEE Transactions on Robotics and Automation, 7(2):193–204, 1991.
21. Hokuyo, Laser range finder Hokuyo URG-04LX. [Online] Avaible at: http://www.hokuyo-aut.jp/02sensor/07scanner/urg.html, consulted on April 2009.
22. Brainstormers, Tribots, "Brainstormers, Tribots Homepage" [Online] Available at: http:\\www.ni.uos.de\index.php?id=2, consulted on April 2009.