Analysis by Synthesis, a Novel Method in Mobile Robot Self-Localization

Alireza Fadaei Tehrani[1, 2], Raúl Rojas[4], Hamid Reza Moballegh[1, 3], Iraj Hosseini[1, 3], and Pooyan Amini[1, 3]

[1] Robotic Center of Isfahan University of Technology
http://robocup.iut.ac.ir
[2] Mechanical Engineering Department, Isfahan University of Technology (IUT)
mcjaft@cc.iut.ac.ir
[3] Electrical Engineering Department, Isfahan University of Technology (IUT)
iraj@sarv-net.com
{moballegh, pooyan_amini}@yahoo.com
[4] Freie Universität Berlin, Takustraße 9, 14195 Berlin, Germany
rojas@inf.fu-berlin.de

Abstract. Fast and accurate self-localization is one of the most important problems in autonomous mobile robots. In this paper, an analysis by synthesis method is presented for optimizing the self-localization procedure. In the synthesis phase of this method, the robot's observation of the field is predicted using the results of odometry. It is done by calculating the position of the landmarks on the captured image. In the analysis phase, the local search algorithms find the exact position of the landmarks on the image from which the best matching coordinates of the robot are determined using a likelihood function. The final coordinates of the robot are then obtained from the odometry sensor, using an integrated delay compensation and correction technique. Experimental results show that precise and delay-free results are achieved with a very low computational cost.

1 Introduction

Self-localization is one of the most challenging problems in the field of autonomous mobile robots. Accuracy of the obtained localization data directly affects the quality of the mission to be done by the robots. In most cases, self-localization must be done in a dynamic, uncertain environment containing a great amount of noise. Such an environment leads to use reliable and precise sensors for data acquisition.

The common sensors used in this field are omni-directional and frontal cameras [3] [13], odometry [4] [7], ultrasonic sensors [6] [12], laser range finders (LRF) [10] [19], and infrared (IR) field surface detectors. Among these, the vision and odometry-based self-localization techniques are currently implemented in most mobile robotic projects [3] [4].

Vision-based self-localization suffers from several disadvantages such as high computational cost, low sample rate, unacceptable delay and inadequate accuracy. According to this, vision techniques are currently used together with other

D. Nardi et al. (Eds.): RoboCup 2004, LNAI 3276, pp. 586 – 593, 2005.

self-localization techniques such as odometry [3]. Several techniques have been developed to fuse odometry and vision outputs together in order to gain more precise and real time results, such as Kalman filter, Markov and Monte Carlo Localization (MCL) [17] [18] [19], and Complementary filtering [20]. In these techniques, vision and odometry algorithms run independently and another module is responsible for fusing their results and calculating the final position and orientation.

In this paper we have introduced a new approach to perform data fusion. In this approach, one of the sensors (usually the simpler sensor) is used to predict another one. This is called the synthesis phase. The position of the landmarks which appear on the captured image is predicted based on the results of the odometry sensor. Having this information, a fast local search can be performed in order to determine the exact position of the landmarks. This is similar to tracking methods except that in tracking, the search is done around the last position of the landmarks on the image. The search area in tracking techniques needs to become wider as the velocity of the robot increases while in our method the search area around the predicted position remains constant. The results of the search algorithm are then converted to the robot coordinates by a probabilistic map matching technique. An integrated delay compensation and correction method is then implemented to obtain the final coordinates of the robot.

The approach has been tested in the RoboCup scenario in which robot positioning is usually a requirement for successful coordination and overall team behavior. The team PERSIA obtained the 3rd place in the middle size league of RoboCup 2003 Italy.

2 Description of the System

Fig. 1 shows the block diagram of the designed self-localization system. In this method, localization data is provided from two sources: omni-directional camera and the odometry system.

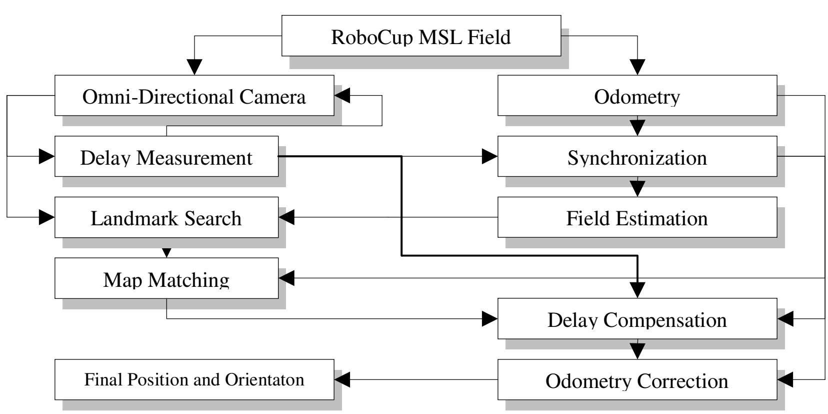

Fig. 1. Block diagram of the self-localization algorithm

The procedure first synchronizes the output of odometry with the camera and then predicts the position of the landmarks on the image based on the synchronized odometric results. Having a preliminary knowledge about the position of the landmarks, the procedure then refers to the image and performs an exact local search to calculate the accurate position of each landmark in the image. The calculated parameters form a local feature map which is matched to the global field map using a likelihood function that is maximized at the most probable coordinates of the robot. Finally an integrated delay compensation and correction technique is used to synchronize the two sources and also eliminate the delay of the final output. Further parts of this section describe the function of each module in detail.

2.1 Odometry

An approach to decrease the cumulative errors of the odometry sensors is presented in [7] by separating the driving and odometry wheels which increases precision of the sensor to a great extent. In the present method, odometry provides the preliminary estimation of the robot coordinates which is then improved by the further parts of the system using vision-based techniques. It is also used to eliminate the delay caused by the vision module.

2.2 Omni-Directional Camera

In many systems, vision is used as a proper complement to cooperate with odometry. However, the vision-based localization is robust enough to be implemented individually. The reason for fusing odometry and vision is high amount of delay, low precision and heavy process of the vision-based techniques.

Omni-directional cameras are preferred to frontal cameras because they can provide the robot with 360° observation range [13] [16].

Another shortcoming of the vision-based techniques is their delay, which is generated partly by the camera and its interface hardware and driver and partly by image processing algorithms. A high amount of delay can easily cause the control system that uses the localization data as its feedback to diverge. Our vision hardware is composed of a camera and a hyperbolic mirror. The delay of such system is measured to be around 100 milliseconds.

2.3 Field Estimation

After synchronizing the odometry results with the current captured frame, the next step towards linking these two sources is to predict the position of the landmarks on the image, based on the odometry generated coordinates. This can efficiently decrease the amount of needed computations, by avoiding the search algorithms from searching a large area of the image which contains no useful landmarks. This also simplifies the search algorithm for each specific landmark because the search area rarely contains other landmarks to be confused with the desired one.

The function that maps the surface of the field to the captured image is a radial transform. Under this transform, the viewing angle of the objects around the robot remains unchanged on the image, while their distance from the center of the image is not linearly related to its real value. Also, every line which is perpendicular to the field surface appears as a radial line in the captured image. According to this phenomenon, the field can be predicted as a set of distances and angles which describe each landmark as it appears in the polar coordinate system placed on the center of the image. The following objects are defined as landmarks; goal vertical bars, intersections between goal inner corners and the field surface, and poles.

The field prediction function is not only used to predict the position of the landmarks in the image, but also it forms the global feature map of the field in the map matching procedure.

2.4 Landmark Detection Algorithms

It is very complicated to search the image for a landmark while having no knowledge of its position because almost every landmark in the MSL field consists of blue and yellow parts and can not be simply distinguished from others according to its color.

Another problem which occurs during the landmark detection is the possibility of having a landmark occluded by another landmark or robots. For example the poles may be occluded by the goal sides from some observation points inside the field.

These problems can be solved by performing a local search around the result of field prediction procedure for each landmark. According to this predicted position, a proper search region can be defined for the detection algorithm that is not related to the velocity of the robot. For most of the landmarks, the search region contains no other landmarks. The region sometimes contains part of the neighbor landmark, so the detection algorithm in the worst case should only distinguish between two neighbor landmarks. This will result in further simplification of the algorithms which indeed, efficiently decreases their computational cost.

2.5 Map Matching

After finding the exact position of each landmark on the image, another module is needed to calculate the position and orientation of the robot. Two methods for performing this task are discussed in this section.

In the first method, the robot coordinates are calculated by triangulation as in [8]. One of the drawbacks of the triangulation method is the high sensitivity of the algorithm to incorrect input parameters. The reason for this sensitivity is that the triangulation algorithms use no means of redundancy in their calculations. Obviously, the robot can localize itself by knowing the position of at least 3 landmarks on the image. The triangulation methods typically use 3-5 parameters to calculate the robot coordinates while more than 15 parameters can be easily derived from the image.

The second method is the probabilistic map matching, which is designed to take advantage of redundancies in the input parameters to reduce the sensitivity of the algorithm to incorrect inputs which may be generated due to noise and occlusion. In this method the best match between the landmark detection results and the global field map is found using maximum likelihood estimation.

In order to compare the feature parameters in the global field map with the detected parameters, a maximum likelihood similarity measure is defined. The function must be designed so that it is not affected seriously if one or more parameters are presented incorrectly. Such a function can be composed as (1) in which S and S' are vectors of the measured and estimated features (angles and distances) and $N_i(.)$ is the i^{th} normal function defined in (2) for comparing corresponding elements of S_i and S'_i. α_i presents the acceptable error range in the compared parameters. Using trial and error, the proper α_i for distances is found to be 0.02 and for angles is found to be 0.1.

$$L(S,S') = \sum N_i(S_i - S'_i) \tag{1}$$

$$N_i(x) = e^{-\alpha_i x^2} \tag{2}$$

Fig. 2 shows the result of applying the likelihood function for comparing a set of features calculated from a typical point with the global field map.

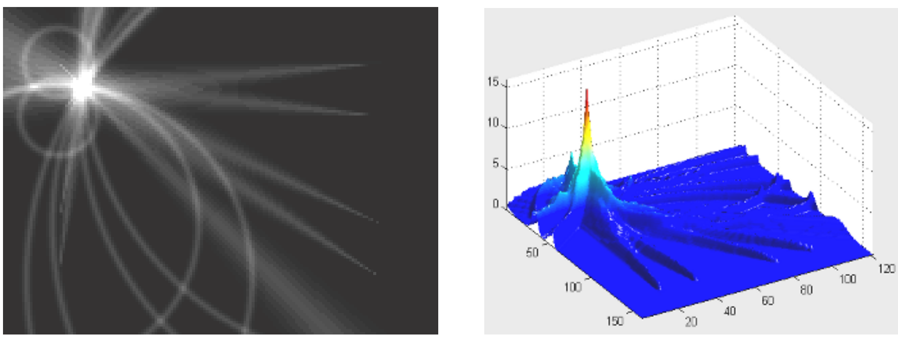

Fig. 2. The likelihood function is used to find the position of the robot

Applying the maximum likelihood measure to all possible coordinates of the robot will need a great amount of computations. In order to decrease the computational cost of the algorithm, the matching can be done in a neighborhood of the odometric coordinates of the robot.

2.6 Global Search

The final coordinates of the robot in the introduced algorithm are the vision-improved version of the odometric coordinates. Since odometry is only capable of tracking the displacement and rotation of the robot, the system must be initialized once with the correct coordinates in order to be able to localize the robot. In this case the local landmark detection algorithms are temporarily substituted with a global detection algorithm for a limited number of landmarks. It is known from the features of omni-directional cameras that both goals can always be viewed in a region which is restricted between two circles. A good approach is to detect the vertical goal bars on a circle inside this area, using an edge detector that finds the edges of the blue and yellow regions.

3 Experimental Results

In order to verify the improvement of the technique introduced in this article, we have compared several features of the analysis by synthesis technique with the previous version of the self-localization algorithm used by the robots. This algorithm is similar to many of the current algorithms in its performance, accuracy and computational cost. In this algorithm, the landmark detection is done by region growing and the coordinates of the robot are calculated using the triangulation method. Both algorithms have run on a 1.2 GHz Pentium III notebook. In the following sections two test procedures to compare accuracy and computational cost of the algorithms are introduced.

3.1 Accuracy

The accuracy of the self-localization systems can be compared based on two parameters; the average of error and its standard deviation. The standard deviation of the localization error is important because in many systems, a medium constant error is more acceptable than a lower value but rapidly varying one. The average values and standard deviation of position and orientation errors are given in Table 1.

Table 1. The average and standard deviation of the localization errors

	Position Err. Avg. (cm)	Position Err. SD (cm)	Orientation Err. Avg. (deg)	Orientation Err. SD (deg)
Old Method	**24**	**12**	**5.1**	**2.4**
New Method	**3.2**	**0.7**	**1.5**	**0.3**

3.2 Computational Cost

Computational cost of self-localization algorithms is the most important factor in implementation of the mobile robots because these algorithms are computationally much more expensive than the other algorithms running on the CPU of the robot. The analysis by synthesis method has reduced this usage to a great amount. In order to compare the computational cost of the algorithms the performance indicator of Windows 2000 is used.

Another feature of the algorithms which should be compared is the constancy of their amount of computations during the game. This feature can be measured by determining the CPU usage of both algorithms when localizing the robot in different positions inside the field. A comparison of CPU usage between the algorithms is made in Table 2. Three typical points inside the field are selected for this comparison, which are center of the field, inside the goals and near the poles.

Table 2. Comparison of CPU usage between new and old methods

	Center of the Field	Inside the Goals	Near the Poles
Old Method	**53%**	**82%**	**64%**
New Method	**39%**	**36%**	**42%**

4 Conclusion

Joint vision-odometry self-localization of moving robots has attracted the attention of many researchers. In this work, the performance of the mobile robot self-localization is improved by proposing a new method called analysis by synthesis. In this method, the vision sensor of the system is first predicted (synthesized) using the odometric coordinates of the robot. The image is then analyzed based on the preliminary prediction to achieve more accuracy in the robot coordinates. An integrated correction and delay compensation module then corrects the coordinates which are obtained from odometry sensor and obtains the final coordinates of the robot. This method has not only reduced the computational cost of the self-localization algorithm but also increased the precision and reliability of the outputs. Experimental results clearly verify the improvements.

References

1. P. MacKenzie and G. Dudek, Precise Positioning Using Model-Based Maps, *Proceedings IEEE International Conference on Robotics and Automation,* San Diego, CA, May 1994.
2. C. Marques, P. Lima, A Localization Method for a Soccer Robot Using a Vision-Based Omni-Directional Sensor, *The 4rth International Workshop on RoboCup,* 2000.
3. F. Marando, M. Piaggio and A. Scalzo, Real Time Self Localization Using a Single Frontal Camera, *The 9th International Symposium on Intelligent Robotic Systems (SIRS 2001),* LAAS-CNRS, Toulouse, France, July 2001.
4. A. Motomura, T. Matsuoka and T. Hasegawa, Self-Localization Method Using Two Landmarks and Dead Reckoning for Autonomous Mobile Soccer Robots, *The 7th International RoboCup Symposium,* Padua, Italy, July 2003.
5. F. V. Hundelshausen and R. Rojas, Tracking Regions, *The 7th International RoboCup Symposium,* Padua, Italy, July 2003.
6. C-C. Tasi, A Localization System of a Mobile Robot by Fusing Dead-Reckoning and Ultrasonic Measurements, *IEEE Transaction on Instrumentation and Measurement,* Vol.47, No.5, pp. 1399-1404, 1998.
7. A. Fadaei Tehrani, A. M. Doosthosseini, H. R. Moballegh, P. Amini, M. M. Daneshpanah, A New Odometry System to Reduce Asymmetric Errors for Omnidirectional Mobile Robots, *The 7th International RoboCup Symposium,* Padua, Italy, July 2003.
8. J. Ji, G. Indiveri, P. Ploeger, A. Bredenfeld, An Omni-Vision based Self-Localization Method for Soccer Robot, *Proceedings of the IEEE IV2003, Intelligent Vehicles Symposium*, Columbus, Ohio, USA, June 2003.
9. H. Utz, A. Neubeck, G. Mayer and G. Kraetzschmar, Improving Vision Based Self-Localization, *The 6th International RoboCup Symposium,* Fukuoka, Japan, June 2002.
10. L. Žalud, L. Kopečný, T. Neužil, Laser Proximity Scanner Correlation Based Method for Cooperative Localization and Map Building, *The 7th International Workshop on Advanced Motion Control (AMC'02),* Maribor, Slovenia, July 2002.
11. C. F. Olson, Probabilistic Self-Localization for Mobile Robots, *IEEE Transactions on Robotics and Automation,* Vol. 16, No. 1, pp. 55-66, February 2000.
12. A. Ohya, A. Kosaka and A. Kak, Vision-Based Navigation by a Mobile Robot with Obstacle Avoidance Using Single-Camera Vision and Ultrasonic Sensing, *IEEE Transaction on Robotics and Automation,* Vol. 14, No. 6, pp. 969-978, December 1998.

13. C. F. Marques, P. U. Lima, Vision-Based Self-Localization for Soccer Robots, *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000),* Vol. 2, pp. 1193-1198, 2000.
14. M. Schmitt, M. Rous, A. Matsikis, K.-F. Kraiss, Vision-based Self-Localization of a Mobile Robot Using a Virtual Environment, *Proceedings of the 1999 IEEE International Conference on Robotics & Automation (ICRA 1999),* Detroit, Michigan, pp. 2911-2916, May 1999.
15. F. V. Hundelshausen, M. Schreiber, F. Wiesel, A. Liers and R. Rojas, MATRIX: A force field pattern matching method for mobile robots, Technical Report B-08-03, *Free University of Berlin*, June 2003.
16. A. Bonarini, P. Aliverti and M. Lucioni, An Omnidirectional Vision Sensor for Fast Tracking for Mobile Robots, *IEEE Transaction on Instrumentation and Measurement,* Vol. 49, No. 3, pp. 509-512, June 2000.
17. J. Wolf and A. Pinz, Particle Filter for Self Localization using Panoramic Vision, *The 27th Workshop of the Austrian Association for Pattern Recognition (OEAGM 2003),* June 2003
18. J.-S. Gutmann, D. Fox, An Experimental Comparison of Localization Methods Continued, *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems,* pp. 454-459, EFPL, Lausanne, Switzerland, October 2002.
19. J.-S. Gutmann, T. Weigel, B. Nebel, A Fast, Accurate, and Robust Method for Self-Localization in Polygonal Environments Using Laser-Range-Finders, *Advanced Robotics Journal,* Vol. 14, No. 8, pp.651-668, 2001
20. S. You and U. Neumann, Fusion of Vision and Gyro Tracking for Robust Augmented Reality Registration, *Proceedings IEEE International Conference on Virtual Reality,* pp. 71-78, Yokohama, Japan, March 2001.