

Single landmark based self-localization of mobile robots

Abdul Bais^{††*}, Robert Sablatnig[§] and Jason Gu[†]

[‡]Institute of Computer Technology
Vienna University of Technology, Vienna, Austria

[§]Pattern Recognition and Image Processing Group
Institute of Computer Aided Automation
Vienna University of Technology, Vienna, Austria

[†]Robotics Research Group
Department of Electrical and Computer Engineering
Dalhousie University
Halifax, Canada

E-mail: bais@ieee.org, sab@prip.tuwien.ac.at, jason.gu@dal.ca

Abstract

In this paper we discuss landmark based absolute localization of tiny autonomous mobile robots in a known environment. Landmark features are naturally occurring as it is not allowed to modify the environment with special navigational aids. These features are sparse in our application domain and are frequently occluded by other robots. This makes simultaneous acquisition of two or more landmarks difficult. Therefore, we propose a system that requires a single landmark feature. The algorithm is based on range measurement of a single landmark from two arbitrary points whose displacement can be measured using dead-reckoning sensors. Range estimation is done with a stereo vision system. Simulation results show that the robot can localize itself if it can estimate range of the same landmark from two different position and if the displacement between the two position is known.

1. Introduction

For reliable navigation in any environment, a mobile robot must know its pose (position and orientation). Therefore, estimating the position of a robot is one of the fundamental problems of mobile robotics. The task can be divided into two sub-tasks: global position estimation and local position tracking [7]. Global position estimation is the

ability to determine the robot's position in an a priori known or previously learned map, given no information other than that the robot is somewhere on the map. On the other hand in local position tracking the robot tracks its position with respect to a known location. Methods for local position tracking suffer from accumulation of minute measurements to obtain the final estimate, whereas, techniques for global position estimation, are less accurate and often require significantly more computational power [7]. This leads to techniques where local measurements are fused with measurements from the robot environment [1, 2, 10, 11, 13, 15, 23]. However, the robot must be able to estimate its position from the very beginning or when/if it loses track of its position during navigation.

The test-bed of our algorithm is a soccer playing autonomous robot, called Tinyphoon¹ [17] (Fig. 1). Currently, soccer robots of the size comparable to that of Tinyphoon use global vision, which refers to the use of camera(s) placed at fixed locations in the workspace to extend the local sensing available on board of each robot. In such type of method, characteristic points forming a pattern on robot are identified and localized by analyzing images taken by the global camera. Robots in the Robocup F180 (www.robocup.org) league are marked with special markers [5]. These markers are tracked and localized in an image taken by a global camera [22]. Similarly, the FIRA Mirobot (www.fira.net) robots are marked with special color patterns so that they can be localized with the help of images taken

*Supported by Higher Education Commission of Pakistan.

¹www.tinyphoon.com

by a global camera. A host computer is used for processing instead of onboard cameras and local processing. In such an approach one can use high speed off-the-shelf computers for processing.



Figure 1. Tinyphoon

If it is not allowed to modify the robot environment with artificial landmarks or active beacons, the robot has to extract naturally occurring distinctive features in its environment to estimate its position with respect to them. Such type of features are very scarce in our application domain and are frequently occluded by other robots for longer durations. This makes simultaneous acquisition of multiple landmarks difficult. As such the localization algorithm should be based on as few landmarks as possible.

Range is an important factor in landmark based methods. If the robot can only measure angles between landmarks then a minimum of three distinct landmarks are required to triangulate the robot position on a planar surface [20, 24]. Furthermore, methods based on angle measurements are sensitive to the relative position of robot with respect to the landmarks [21]. Whereas, when range measurements are available this requirement drops to two if ordering of the landmarks with respect to the robot is possible [3].

There have been approaches to maximize the chances of simultaneous acquisition of multiple landmarks using omnidirectional cameras with viewing angle of 360° [12, 18]. A camera looking at a mirror of special shape provides the robot with visual information in all directions simultaneously. The work reported in [14] is an example of such a setup used for localization of soccer robots. As the resolution and camera range depend on the mirror geometry, the resolution of objects is usually smaller than that of directional images.

Approaches using single frontal cameras in conjunction

with odometry are widely used for self-localization. These methods are either based on calculating range and/or bearing based on known shape and size of landmarks or enforce special constraints on environment features [6, 9]. Choi et al. [9] use calibrated camera to acquire depth of features. The robot's actual position is computed by referencing door corners over the floor edge, and heading by direction of the floor edge lines. Single image based range calculation is erroneous and the approach cannot be applied all the time [16].

Omnidirectional cameras with viewing angle of 360° provide more landmarks but suffer from high cost of the mirror, low resolution of the camera, and requirement of an additional space to fit the mirror and the camera. With frontal cameras one can have high resolution but the field of view is limited. Furthermore, range measurement using single image is too erroneous and the approach cannot be used all the time [16].

The localization algorithm uses the concept of *runningfix* [8]. The underlying principle of the running fix is that an angle or range obtained from a landmark at time $t - 1$ can be utilized at time t , as long as the cumulative movement vector, recorded since the reading was obtained, is added to the position vector of the landmark, thus creating a virtual landmark. In [8], they discuss bearing estimation with an infra red beacon/detector pair, whereas we conduct experiments using stereo range estimation to a natural landmark.

The balance of the paper is organized as follows: Section 2 discusses robot position estimation based on range measurement of a distinct landmark from two different points with known displacement. Potential landmarks and their extraction is analyzed in Section 3. Experimental results are presented in Section 4, finally the paper is concluded in Section 5.

2. Calculating Robot Position

We assume that the robot's motion is two dimensional where the robot pose has 3 degrees of freedom i.e. x , y and θ . As can be seen in Fig. 2, the global coordinate system is represented by X and Y axis and the robot coordinate system by x_r and y_r axis. Rotation of robot coordinate system with respect to the global coordinate system is represented by the angle θ . Suppose the robot detects a distinct landmark point p_l at (x_l, y_l) in the global coordinate system and measures its distance r_1 . This constrains the robot's position to a circle C with radius r_1 and center at the landmark location. This circle, as shown in Fig. 3, is described by (1).

$$(x - x_l)^2 + (y - y_l)^2 = r_1^2 \quad (1)$$

After detecting this landmark the robot tries to identify other landmarks. In case this is the only landmark that can

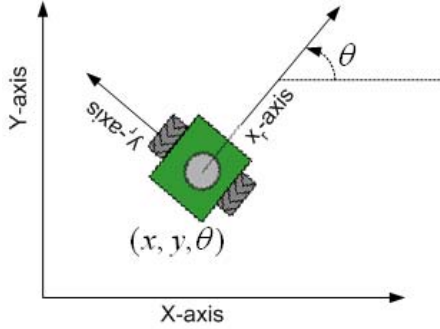


Figure 2. Robot position in the global frame of reference

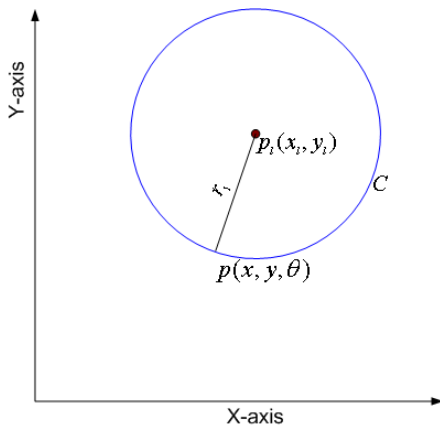


Figure 3. Measuring range to a single distinct landmark constrains the robot position to a single circle

be detected from its current position, the robot travels to a new point in its environment and tries to detect the same landmark again. After detecting landmark point p_l the robot knows that it is somewhere on circle C . Movement of the robot by d_x and d_y in the X and Y directions can be seen as motion of the circle C by the same amount. Radius of the circle remains equal to r_1 as all candidate positions undergo the same displacement. This moment of the circle can be described as if the circle's center has moved to a new location, namely, x'_l and y'_l (Fig. 4). Taking this change into consideration (1) can be reformulated as in (2). Now the robot is somewhere on circle C' . We may think of (x'_l, y'_l) as a virtual landmark. The original circle C is shown dotted.

$$(x - x'_l)^2 + (y - y'_l)^2 = r_1^2 \quad (2)$$

where

$$\begin{aligned} x'_l &= x_l + d_x \\ y'_l &= y_l + d_y \end{aligned}$$

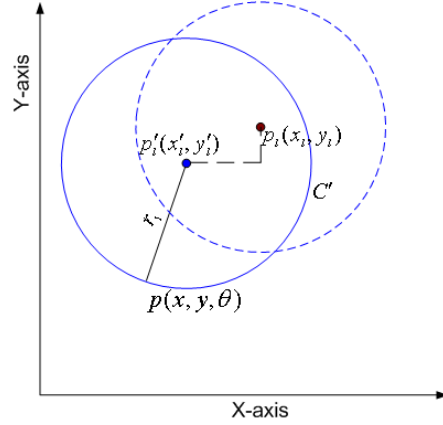


Figure 4. Robot's movement is effectively motion of the circle by the same amount

Suppose the robot is able to detect the same landmark p_l from this new location and measures its range r_2 . This measurement constrains the robot position to a new circle, C'' , which is described by (3).

$$(x - x_l)^2 + (y - y_l)^2 = r_2^2 \quad (3)$$

As illustrated in Fig 5 and from the solution of (2) and (3), the intersection of circle C' and C'' will give possible robot position in the global coordinate system. Subtracting (3) from (2) and re-arranging terms we have

$$x = A + By \quad (4)$$

where

$$\begin{aligned} A &= \frac{r_1^2 - r_2^2 + x_l^2 - x'_l{}^2 + y_l^2 - y'_l{}^2}{2(x_l - x'_l)} \\ B &= \frac{y'_l - y_l}{x_l - x'_l} \end{aligned}$$

Substitution (4) in (3) results in

$$Cy^2 + Dy + E = 0 \quad (5)$$

where

$$\begin{aligned} C &= B^2 + 1 \\ D &= 2AB - 2Bx_l - 2y_l \\ E &= A^2 + x_l^2 - 2Ax_l + y_l^2 - r_2^2 \end{aligned}$$

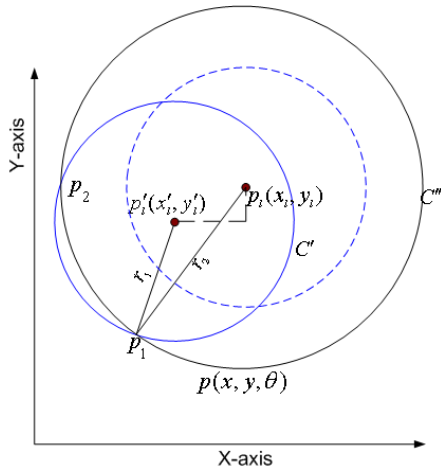


Figure 5. Position of the robot is constrained to the intersection of circle C' and C''

Solving (5) using quadratic formula we have

$$y = \frac{-D \pm \sqrt{D^2 - 4CE}}{2C} \quad (6)$$

One of the solution pairs (p_{x_1}, p_{y_1}) and (p_{x_2}, p_{y_2}) from (6) and (4) will qualify for the possible robot position. The ambiguity between the two possible positions of the robot is resolved by a rough estimate of the robot position or by tracking both candidate positions until further information is available. As measurements are not error free, existence of a correct solution pair depends on error in range measurement, error in landmark position and the displacement vector. Nevertheless, now the robot position belief is concentrated to two points from a state where the robot is totally unaware of its position.

3. Landmarks for self-localization

We use color transitions, corners, junctions and line intersections as landmarks. Corners, junctions and line intersection are detected using semantic interpretation of line segments extracted using gradient based Hough transform [4]. The gradient based Hough transform provides the strongest groupings of collinear pixels having roughly the same edge orientation. Groups of pixels are then processed to calculate length and end points of line segments. This is followed by classification of field markings which are bright lines and arcs on a dark background forming double edges of opposite gradient. Fig. 6 illustrates the detection of these features.

The vertical edges of the goal corners are normally missed during edge detection and subsequent line segments extraction as the change in y-channel value between white

and yellow is not significant and the length of the edge is small as compared to other lines in the environment. Therefore, we extract goal corners based on color transitions as discussed in the following section [3].

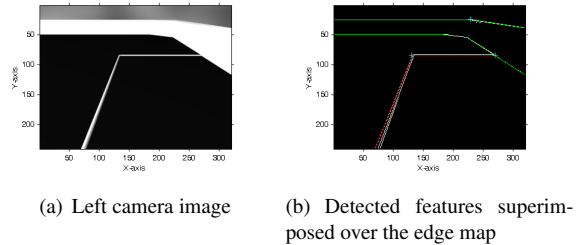


Figure 6. Line based landmarks for self-localization

3.1. Detecting goal corners

Goals are marked with different colors (blue and yellow). We use color segmentation of the camera images to detect corners of the goal. The process is outlined as follows.

In the left camera image, pixels are tested if they belong to either blue or yellow color. This 'segmentation' is done at a lower scale. Every fourth pixel in a row of every fourth row is tested, which results in a rectangular window (s) around color patches, if any. If a blue or yellow color patch is found in the image, the neighborhood of the rectangular window is searched for color transition (color transition from white to yellow, yellow to white, white to blue, or blue to white represents a goal corner), at full scale. If a color transition is detected in the left image, the corresponding feature points are searched in the right image. The search in the right image is based on parameters of the feature points in the left image. If the corresponding feature point is detected in the right image, its distance from the current robot position is calculated.

The use of two colors (white and blue/yellow) to detect a transition makes the process robust against outliers. All rows inside the rectangular window are searched for transition pixels. One value in a group of pixels is taken as the x-component of the edge between the wall and the colored goal. Outliers in the group are eliminated using simple statistical measures.

3.2. Calculating robot orientation

In this section we discuss calculation of robot orientation with respect to goal corners which is done after position is estimated. The process is illustrated in Fig. 7. When robot calculates its position with respect to a yellow goal corner,

θ can be calculated using (7).

$$\theta = \alpha_1 - \alpha_2 \quad (7)$$

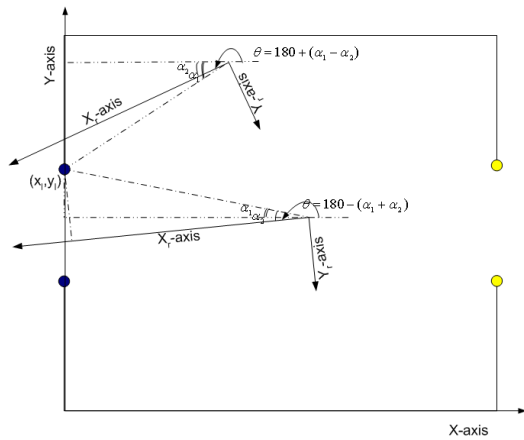
where $\alpha_1 = \arctan\left(\frac{y_l - p_y}{x_l - p_x}\right)$ and $\alpha_2 = \arctan\left(\frac{y_r}{x_r}\right)$.

In these equations (x_l, y_l) is the location of the landmark, (p_x, p_y) is robot position and (x_r, y_r) is the landmark location in robot coordinate system. The landmark and robot position are in global coordinate system.

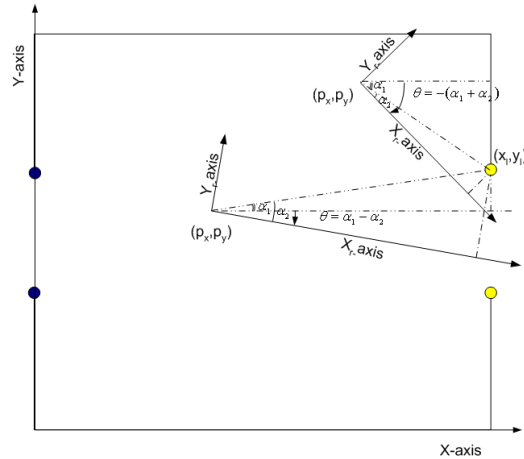
Similarly when the robot position is calculated with respect to a blue goal corner θ can be calculated using (8) as shown in Fig 7(b).

$$\theta = 180^\circ - (\alpha_1 + \alpha_2) \quad (8)$$

where $\alpha_1 = \arctan\left(\frac{l_y - p_y}{p_x}\right)$ and $\alpha_2 = \arctan\left(\frac{y_r}{x_r}\right)$.



(a) Blue goal



(b) Yellow goal

Figure 7. Robot orientation with respect to goal corners

4. Experimental Results

We simulate the performance of the method on synthetic data. In this study we use only four landmarks that correspond to the goal corners. These landmarks are detected as color transition from white to blue/yellow or vice versa. In total 500 different trials were conducted. In each of these trials the robot is given two arbitrary points. Starting at the first point, the robot takes images of its surrounding environment with its stereo vision system. These images are processed to extract color transition (if any). If a color transition(s) is found, its depth is estimated. In case there is only one landmark detected the robot travels to the second point and repeats the process. If the same landmark is also observed from the second location, then these range measurements, location of the landmark in the global coordinate system and displacement vector of the robot is used to estimate position as discussed in Section 2. However, depending on the given points, the robot may not be able to detect the same landmark from both the locations or the range measurement error may be such that the two circles don't intersect each other.

The size of the robot and dimension of the field conforms with the FIRA MiroSot Small League². The only change is that the two goals are in different color. The baseline of the stereo vision system is 3 cm. This comes from the construction of the robot as the maximum allowed size of the robot is 7.5cm. Image resolution can be set at $(320 \times 240, \text{QVGA})$ or $(640 \times 480, \text{VGA})$. First we conduct our experiment with image resolution of (320×240) pixels and then repeat the same with high image resolution (640×480) to study the effect of quantization error. In both these experiments image resolution is the only change.

Statistical results for error in pose are shown in Table 1 and Table 2 for both of these experiments, while range measurement error is shown in Table 3 and Table 4. The first column in all tables show values for the mean error. The standard deviation (Std), minimum (Min) and maximum (Max) values for each group are presented in the 2nd, 3rd and 4th columns. Errors δx and δy in Table 1 and Table 2 and the absolute error in Table 3 and Table 4 is expressed in centimeters. The normalized error is the absolute error in range measurement divided by the actual range.

For the low resolution case we have conducted a total of 4206 range measurements. Histograms for absolute and normalized range measurement error are shown in Fig. 8(a) and Fig. 8(b), respectively. Whereas, for the high resolution we have only 968 range measurements. Histograms for this case are shown in Fig. 9(a) and Fig. 9(b). In these figures, numbers along the x-axis represent the absolute (millimeters) and normalized (%) range measurement error,

²<http://www.fira.net>

Table 1. Error in x, y and θ , image resolution (320×240) pixels

	Mean	Std	Min	Max
δx	10.03	9.98	0.02	44.48
δy	16.91	15.59	0.002	68.94
$\delta \theta$	14.60°	11.76°	0.001°	66.52°

Table 2. Error in x, y and θ , image resolution (640×480) pixels

	Mean	Std	Min	Max
δx	8.85	6.26	0.03	29.32
δy	12.74	10.49	0.08	57.30
$\delta \theta$	12.26°	7.63°	0.08°	36.98°

Table 3. Range measurement error, image resolution (320×240) pixels

	Mean	Std	Min	Max
Absolute	4.89	4.23	0.001	30.81
Normalized	5.74%	4.28%	0.001%	35.91%

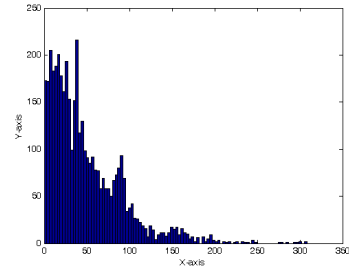
Table 4. Range measurement error, image resolution (640×480) pixels

	Mean	Std	Min	Max
Absolute	4.50	2.83	0.12	20.74
Normalized	5.07%	2.07%	0.10%	13.94%

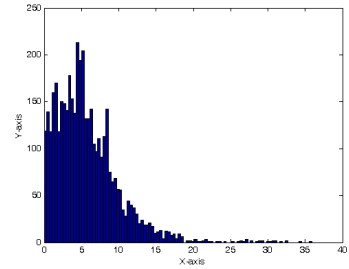
whereas, those along the y-axis show the counts for the corresponding bin.

It is clear from the error statistics that better results can be achieved with high resolution images. For the low resolution case, the robot position is estimated only in 176 trials where this number is 381 when high resolution images were used. Another option is to use sub-pixel stereo. Both these options require extra processing power, which is not always available. Error histograms for both the experiments are shown in Fig. 10 and Fig. 11. δx and δy are in millimeters whereas $\delta \theta$ is in degrees.

Low resolution is not the only reason for high range estimation error. We use a very narrow baseline stereo (3cm) since the construction of the robot does not allow the use of wide baseline. Moreover, due to the size and concavity of the goal, it is difficult to determine which point on the goal is being observed. This results in inconsistent ranges and inconsistent landmark positions [19].

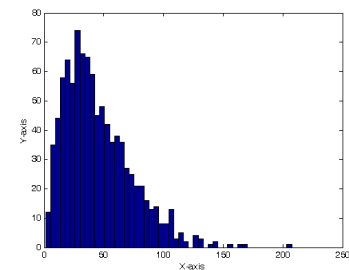


(a) Absolute error

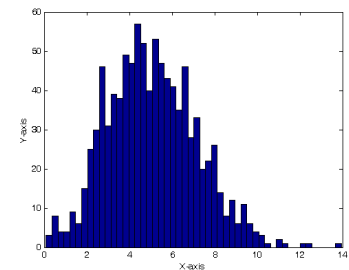


(b) Normalized error

Figure 8. Stereo range measurement error: image size 320×240

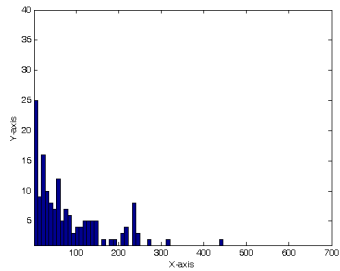


(a) Absolute error

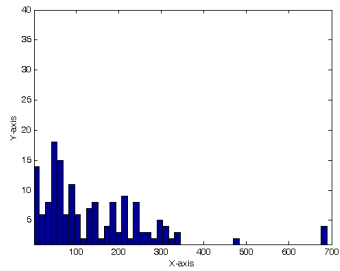


(b) Normalized error

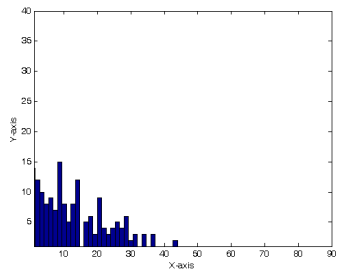
Figure 9. Stereo range measurement error: image size 640×480



(a) δx

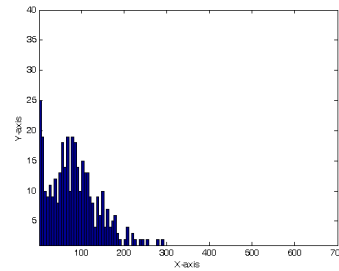


(b) δy

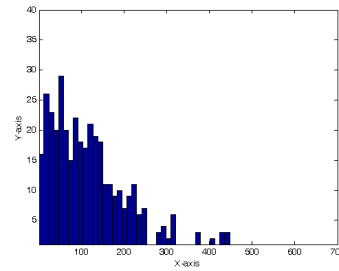


(c) $\delta \theta$

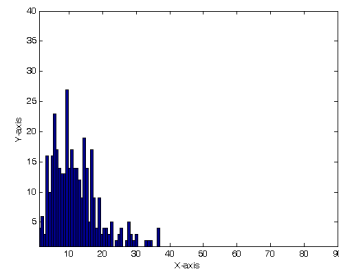
Figure 10. Histograms for pose error: image size 320×240



(a) δx



(b) δy



(c) $\delta \theta$

Figure 11. Histograms for pose error: image size 640×480

5. Conclusion

In this paper we have presented our investigation of self-localization using range measurement to a single landmark from two arbitrary points, where the displacement between the two is known. Simulation results show that this method can successfully localize robots in an environment where landmarks are very scarce. Only four landmarks that correspond to the color transitions are used. The stereo vision system is used to extract landmarks and estimate their depth, whereas the actual displacement of the robot is fed into the algorithm. Detection of landmark is very efficient as only $N/16$ pixels are tested to determine rectangular boundaries around color blobs (if any), N being the total number of pixels [3]. This local area is then searched for color transition. To illustrate the effect of low image reso-

lution we repeat the same experiment with high resolution images (640×480 , VGA). All parameters except image resolution are kept the same.

The main drawback of this method is that it requires the robot to drive further due to high range error. It is very unlikely to estimate correct position if range measurement error is comparable to the robot displacement. This results in erroneous displacement vector as odometric errors accumulate as the robot moves further. Nevertheless the method is able to concentrate the robot position belief to two points where the robot is totally uncertain about of position. Currently we are studying the effect of dead-reckoning error on position estimate, methods for efficient interpretation of landmarks other than color transitions and tracking of landmarks.

References

- [1] G. Adorni, S. Cagnoni, S. Enderle, and G. K. Kraetzschmar. Vision-based localization for mobile robots. *Robotics and Autonomous Systems*, 36:103–119, 2001.
- [2] G. Adorni, S. Cagnoni, and M. Mordonini. Landmark-based robot self-localization: a case study for the robocup goal-keeper. In *Proceedings of the International Conference on Information Intelligence and Systems*, pages 164–171, Bethesda, MD, USA, October 1999.
- [3] A. Bais and R. Sablatnig. Landmark based global self-localization of mobile soccer robots. In P. J. Narayanan, S. K. Nayar, and H.-Y. Shum, editors, *Computer Vision ACCV 2006: 7th Asian Conference on Computer Vision*, volume 3852 of *Lecture Notes in Computer Science*, pages 842–851, Hyderabad, India, January 2006. Springer-Verlag GmbH.
- [4] A. Bais, R. Sablatnig, and G. Novak. Line-based landmark recognition for self-localization of soccer robots. In *IEEE International Conference on Emerging Technologies (ICET '05)*, pages 132–137, Islamabad, Pakistan, September 2005.
- [5] D. Ball, G. Wyeth, and S. Nuske. A global vision system for a robot soccer team. In *Proceedings of the 2004 Australasian Conference on Robotics and Automation*, 2004.
- [6] T. Bandlow, M. Klupsch, R. Hanek, and T. Schmitt. Fast image segmentation, object recognition and localization in a robocup scenario. In *RoboCup-99: Robot Soccer World Cup III*, pages 174–185, 1999.
- [7] J. Borenstein, H. R. Everett, and L. Feng. *Navigating Mobile Robots: Systems and Techniques*. A. K. Peters, Ltd., 1996.
- [8] M. Case. Single landmark navigation by mobile robot. In *Proceedings of the 1986 SPIE Conference on Mobile Robots*, volume 727, pages 231–237, Cambridge, MA, October 1986. This paper discusses the idea of a virtual landmark.
- [9] W. Choi, C. Ryu, and H. Kim. Navigation of a mobile robot using mono-vision and mono-audition. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (IEEE SMC '99)*, volume 4, pages 686–691, 1999.
- [10] H. I. Christensen, N. O. Kirkeby, S. Kristensen, and L. Knudsen. Model-driven vision for in-door navigation. *Robotics and Autonomous Systems*, 12(3-4):199–207, April 1994.
- [11] F. de Jong, J. Caarls, R. Bartelds, and P. Jonker. A two-tiered approach to self-localization. In *RoboCup 2001: Robot Soccer World Cup V*, pages 405–410. Springer-Verlag, 2002.
- [12] J. Ji, G. Indiveri, P. Ploeger, and A. Bredenfeld. An omnivision based self-localization method for soccer robot. In *IEEE symposium on Intelligent Vehicles (IV'03)*, Columbus, Ohio, USA, June 2003.
- [13] L. Iocchi and D. Nardi. Hough localization for mobile robots in polygonal environments. *Robotics and Autonomous Systems*, 40(1):43–58, July 2002.
- [14] C. F. Marques and P. U. Lima. A localization method for a soccer robot using a vision-based omni-directional sensor. In P. S. et al., editor, *RoboCup 2000: Robot Soccer World Cup IV*, number 2109 in LNCS, pages 96–107, 2001.
- [15] A. Motomura, T. Matsuoka, and T. Hasegawa. Self-localization method using two landmarks and dead reckoning for autonomous mobile soccer robots. In *RoboCup 2003: Robot Soccer World Cup VII*, LNCS, pages 526–533, 2003.
- [16] S. B. Nickerson, P. Jasiobedzki, D. Wilkes, M. Jenkin, E. Milios, J. Tsotsos, A. Jepson, and O. N. Bains. The ark project: Autonomous mobile robots for known industrial environments. *Robotics and Autonomous Systems*, 25(1-2):83–104, October 1998.
- [17] G. Novak and S. Mahlknecht. TINYPHOON a tiny autonomous mobile robot. In *IEEE International Symposium on Industrial Electronics (ISIE' 05)*, pages 1533–1538, June 2005.
- [18] G. Steinbauer and H. Bischof. Illumination insensitive robot self-localization using panoramic eigenspaces. In *RoboCup 2004*, volume 3276 of *LNAI*, pages 84–96, 2005.
- [19] A. W. Stroupe, K. Sikorski, and T. Balch. Constraint-based landmark localization. In G. Kaminka, P. Lima, and R. Rojas, editors, *RoboCup 2002: Robot Soccer World Cup IV*, volume 2752 of *LNCS*, pages 8–24. Springer-Verlag, November 2003.
- [20] K. Sugihara. Some location problems for robot navigation using a single camera. *Computer Vision, Graphics, and Image Processing*, 42(1):112–129, 1988.
- [21] K. T. Sutherland and W. B. Thompson. Inexact navigation. In *IEEE International Conference on Robotics and Automation (ICRA' 93)*, pages 1–7, 1993.
- [22] Z. Tang, A. Nazeer, and Z. Sun. Robust vision localization for robocup f180. *Singapore Polytechnic Technical Journal*, May 2003.
- [23] A. F. Tehrani, R. Rojas, H. R. Moballeghe, I. Hosseini, and P. Amini. Analysis by synthesis, a novel method in mobile robot self-localization. In G. K. et al., editor, *RoboCup 2004: Robot Soccer World Cup VIII*, volume 3276 of *LNCS*, pages 586–593, 2005.
- [24] D. C. K. Yuen and B. A. MacDonald. Vision-based localization algorithm based on landmark matching, triangulation, reconstruction, and comparison. *IEEE Transactions on Robotics*, 21(2):217–226, April 2005.