Fuzzy Landmark-Based Localization for a Legged Robot

P. Buschka **A. Saffiotti** **Z. Wasik**

Center for Applied Autonomous Sensor Systems
Dept. of Technology, Örebro University
S-70182 Örebro, Sweden
{par.buschka,alessandro.saffiotti,zbigniew.wasik}@aass.oru.se

Abstract

We describe a new technique for landmark-based self-localization which is suitable for robots with poor odometry. This technique uses fuzzy logic to account for errors and imprecision in visual recognition, and for extreme uncertainty in the estimate of the robot's motion. It only requires an approximate model of the sensor system and a qualitative estimate of the robot's displacement, and it has a moderate computational cost. We show examples of use of our technique on a Sony AIBO legged robot in the RoboCup domain.

1 Introduction

The motivation for this work came from the need to develop an algorithm for reliable self-localization on a legged robot. Although a number of effective self-localization techniques have been proposed in the field of mobile robotics, these are usually geared toward wheeled robots, and their application to legged robots may be problematic.

Most classical localization techniques like Kalman filter perform *position tracking*, that is, they combine information about the previous position of the robot with information about the robot's motion and with perceptual information to obtain a new position estimate. These techniques tend to be computationally efficient, but they may fail dramatically if the robot's estimated position is too far away from the real one. This makes these techniques difficult to apply to legged robots, since the motion data is subject to major errors due, for instance, to leg slippage and to momentary loss of balance.

More recently, a number of techniques for *absolute localization* have been proposed, which keep a global view on the set of possible positions, and are thus able to recover from arbitrarily large errors. One of the most popular techniques of this kind is Markov localization, which integrates absolute localization and position tracking in a uniform framework [6]. While Markov localization has produced impressive results, these results rely on the use of sophisticated sensors and on the availability of accurate and time-invariant models of them. This condition is difficult to guarantee in a legged robot, where the observation conditions depend, among other things, on the 3D posture.

In this paper, we propose a self-localization technique based on fuzzy logic which is suitable to be used on a legged robot equipped with inexpensive sensors. This technique relies on the observation of known landmarks using a camera sensor, and on the integration of the position information derived from these observations into a fuzzy position grid (FPG). The distinctive features of our technique are as follows.

- It only uses qualitative motion information;
- only uses sporadic observations;
- only needs an approximate sensor model;
- can recover for arbitrarily large localization errors; and
- has a limited computational cost.

We have applied our localization technique on a Sony legged robot in the Robocup domain [4, 5] — see Fig. 1. The above features adequately match the requirements of this application, where: (i) egomotion information is extremely inaccurate; (ii) landmarks can only be observed sporadically, since the camera is most of the time used for tracking the ball; (iii) visual landmark recognition is subject to unpredictable errors due to partial occlusion and mislabeling; (iv) the robot can be pushed by other robots in unexpected locations; and (v) the environment is highly dynamic, thus requiring fast real-time operation.

In the next section, we sketch our approach to landmark recognition, and discuss how we model uncertainty using fuzzy sets. The following sections detail

Proc of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS-2000)
Takamatsu, Japan, 2000. pp 1205-1210.

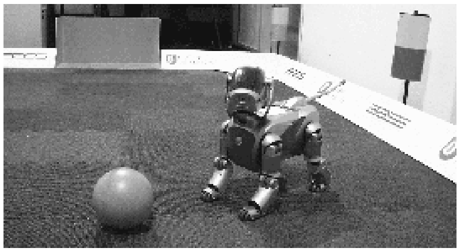


Figure 1: A Sony AIBO robot in the Robocup field. Two visual landmarks can be seen on the right.

our self-localization technique and show a few illustrative examples taken from real data.

2 Landmark acquisition

The considered domain consists of a $2\,m \times 3\,m$ green field with white borders and white lines, two nets, and six cylindric landmarks — see Fig. 1. The nets and the landmarks are uniquely identified by their color, and the geometry of the field is known. Since the nets are typically only partially visible in a crowded field, we use the landmarks and the white lines and borders as our main cues.[1]

To recognize objects, we rely on the color identification hardware in the Sony robot to extract color regions, and use a model-based approach to combine these regions into objects of some type. For instance, a white region adjacent to a green one is taken to be a line object; and a blue region over a pink one forms a landmark object. We then check this object against a number of criteria, which depend on its type. For a landmark, we check that: (1) the object has an acceptable minimum size; (2) the two regions have similar size and horizontal position; (3) the gap and overlap between these regions fall within some given bounds; and (4) the object's position is "reasonable," e.g., a landmark should not lay on the ground.

Once we have identified an object, we estimate its distance and bearing from the robot's axis. In the case of landmarks, we estimate distance based on the size of the object in the image. We build two separate estimates, one from the width and one from the height of the object: Since both may overestimate the real distance in case of partial occlusion, we take the minimum between these two. Bearing is simply estimated from the horizontal position of the object in the image and from the pan angle of the robot's head.

[1]The examples reported in this paper only use landmarks.

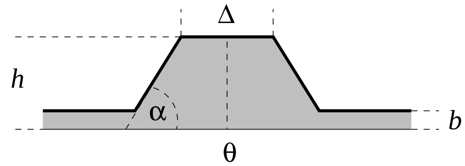


Figure 2: Fuzzy set representation of an angle measurement θ.

An important aspect of our approach is the way to represent the uncertainty in these estimates. There are two main facets of this uncertainty. First, the *imprecision* in the measurement, i.e., the dispersion of the estimated values inside an interval that includes the true value. Imprecision in the distance estimate mainly originates from variations in the size of the object inside the image; these are due to imprecise segmentation, to perspective distortion, and to the rolling of the robot during walking. Imprecision in the angle estimate mainly originates from the play of the head joints, and from the roll of the robot's body. The second aspect is *unreliability*, i.e., the possibility of outliers. False measurements can originate from the partial occlusion of a landmark, from a false identification, or from a mislabeling.

Since the distance and angle measurement are made independently we model their uncertainty separately. Moreover, in order to better separate the impact of imprecision and unreliability, we represent this uncertainty using fuzzy sets under a possibilistic interpretation [11, 9]. For angular information, we use the trapezoidal fuzzy set shown in Fig. 2. The top of trapezoid (*core*) identifies those values which are fully possible: any one of these values could equally be the real one given the inherent imprecision of the measure. The base of the trapezoid (*support*) identifies the area where we could still possibly have meaningful values, i.e., values outside this area are impossible given the measure. In order to account for unreliability, then, we include a small uniform bias b, representing the degree of possibility that the actual angle is "somewhere else" with respect to the measurement. The representation of a distance measurements is similar.

We represent a trapezoidal fuzzy set by the tuple $\langle \theta, \Delta, \alpha, h, b \rangle$, where θ is the center, Δ is the width of the core, α is the slope, h is the height, and b is the bias. In our domain, we have estimated these parameters by visual inspection of data from a limited sample of measurements. However, it is important to note that the technique described below does *not* critically rely on the accuracy of these parameters.

3 Fuzzy self-localization

Our approach to self-localization combines ideas from the fuzzy landmark-based localization proposed by Saffiotti and Wesley [10], and from the position probability grids proposed by Burgard *et al.* [3]. We represent the robot's belief in its position by a possibility distribution over the set of possible positions, represented by a (virtual) 3D fuzzy position grid, or FPG. The information in the grid can represent, and track, multiple possible positions where the robot might be. We give here the basic structure of the algorithm, and give more details in the following sections.

Maintenance of position information follows the typical *predict-observe-update* cycle of recursive state estimators [7]. The *observation* of a landmark at a know position is converted to a possibility distribution on the grid: for each cell in the grid, this measures the possibility of the robot being at that cell given the observation. This possibility distribution is then used to *update* the one representing the prior position estimate by fuzzy intersection. This procedure is repeated for every available observation. Note that the prior distribution can be uniform in case of total ignorance, e.g., when the robot is first placed on the field.

The *predict* part of the cycle is peculiar, since we consider an extremely weak model of the robot's motion. Our prediction simply "blurs" the distribution in the grid to reflect the possibility that the robot has moved somehow. Blurring is done in all directions by an amount that depends on the maximum achievable speed, and it does not diminish the possibility that the robot is still at the same location. No estimate of the actual speed and direction of motion is used in the blurring. This means that our technique is currently odometry free: this was done to see how well we could localize even ignoring egomotion information, which is highly unreliable for a legged robot. Motion information could be included, if desired, by biasing the amount and the direction of blurring according to the estimated direction and speed of motion.

At every moment, the most likely position is given by the center of the region in the grid with the highest possibility. The reliability of this estimate is given by the extent of this region; this information is used, e.g., to decide to engage in a relocalization behavior.

4 Using range information

We first give the details of the above mechanism assuming that only range information is available.

The robot's position at time t is represented by a two-dimensional fuzzy gridmap G_t such that $G_t(x, y)$ measures the degree of possibility, on the $[0, 1]$ scale, that the robot is at position (x, y) in the environment.

To encode the observation of a known landmark at time t, we build a possibility distribution $S_t(\cdot \mid r)$ such that $S_t(x, y \mid r)$ measures the possibility that the robot is at (x, y) given that the observed distance to the landmark is r. The value of $S_t(x, y \mid r)$ depends on the actual distance d from cell (x, y) to the landmark, and is given by the degree of membership of d to the fuzzy set $\langle r, \Delta, \alpha, h\rangle$ associated to r (see Fig. 2).

For each observation, we update the map G_t by

$$G'_t(x, y) = G_t(x, y) \times S_t(x, y \mid r), \qquad (1)$$

where $\times$ denotes a fuzzy intersection operator. There are several choices for $\times$, depending on the independence assumptions that we can make about the items being combined [1]. Since our observations are independent, we use the product operator which reinforces the effect of consonant observations.

For the blurring, we apply an operator borrowed from mathematical morphology and usually applied to image processing: the *dilation* operator $\oplus$. Given a structuring element B that reflects the desired amount of blurring, the dilation of G_t by B is given by

$$G_{t+1} = G_t \oplus B. \qquad (2)$$

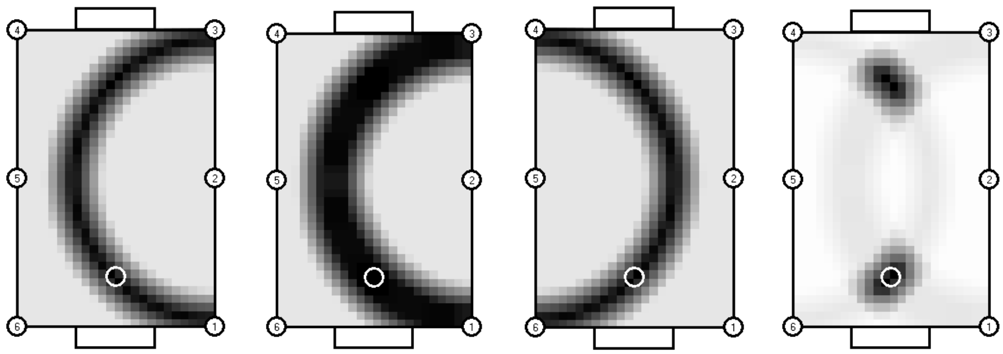
This corresponds to performing a one-step prediction of the robot's position. More precisely, since our position grid can be seen as a fuzzy image, we apply a form of fuzzy dilation [2] given by

$$G_{t+1}(x, y) = \sup_{i,j} \min(G_t(i, j), B(i - x, j - y)). \qquad (3)$$

Since we currently ignore motion information, we use a circular B that blurs the grid equally in all directions. However, motion information can be included in the prediction by using a B that takes the estimated speed and direction of motion into account.

Finally, the most likely position estimate P_t at time t is computed by taking the center of gravity of G_t. The reliability of this estimate is given by the extent of the region in G_t with the highest possibility.

Fig. 3, taken from real data, illustrates our technique. The degree of grey of each cell indicates the possibility of being at that cell. (a) shows G_t after landmark number 2 has been observed. After some time (b), the map has been blurred into G_{t+1} by (2) to account for possible motions. At the same time, landmark 5 is observed, thus producing the distribution $S_{t+1}(x, y \mid r)$ shown in (c). The result of combining S_{t+1} with G_{t+1} by (2) is shown in (d): the effect of the



(a) (b) (c) (d)

Figure 3: Self-localization using only range information. Darker cells have higher possibility. The white circle marks the real position of the robot. The circled numbers mark the position of the six landmarks.

fuzzy intersection is clearly visible, and the new distribution gives a more focused estimate. However, there is still an ambiguity between two possible positions: in the next section, we show how angular information can help resolving these ambiguities.

5 Adding angular information

Since we want to know the orientation of the robot, and not only its position, it is natural to include angular information in our map. A straightforward way to do so is to consider a 3D grid $G_t(x, y, \theta)$, where θ represents the robot orientation. (This is the path followed in [3].) Accordingly, a landmark observation $\vec{r}$ (where $\vec{r}$ is now a vector that includes both range and bearing information) is represented by a possibility distribution S such that $S_t(x, y, \theta \mid \vec{r})$ denotes the possibility that the robot is at (x, y, θ) given the observation $\vec{r}$. Update and blurring are still done according to (1) and (2), respectively, which extend naturally to the 3D case.

This 3D representation has the problem of having very high computation complexity, both in time and space. Given that in our domain all landmarks are uniquely identified, we can simplify the model by removing the ability to handle multiple orientation hypotheses. Instead of representing each possible orientation into a 3D grid, we use a 2D position grid and associate each cell with a pseudo-trapezoidal fuzzy set $\langle \theta, \Delta, \alpha, h, b \rangle$ representing an uncertain orientation (see Fig. 2 above). For a given cell (x, y), this fuzzy set can be seen as a compact representation of a possibility distribution over the cells $\{(x, y, \theta) \mid \theta \in [-\pi, \pi]\}$ of a 3D grid. Accordingly, we have to modify our intersection and dilation operators to operate on this representation of the angle dimension.

To intersect two direction fuzzy sets A and B into C, we first check if A and B overlap — see Fig. 4. If they do (top), we let C be the inner trapezoidal envelope of the real fuzzy intersection: note that the parameters of the new trapezoid can be easily computed from the parameters of A and B. The new bias, in particular, is given by: $b_C = b_A \cdot b_B$.

If A and B do not overlap, i.e., if the new observation and the current estimate are incompatible. then the above computation is not adequate since it would only preserve the shape of one of the two trapezoids, possibly losing any memory of past observations. In this case, we let C be the outer trapezoidal envelope of the fuzzy union of A and B (bottom of Fig. 4). This increases both the width (imprecision) of the estimate, and its bias (unreliability), which is a result to be expected from having observed contradictory information. Notice that this operator is not associative, which means that we might get different results depending on the order in which observations are made.

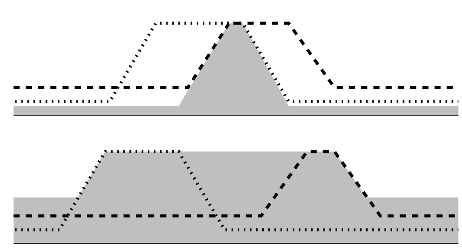
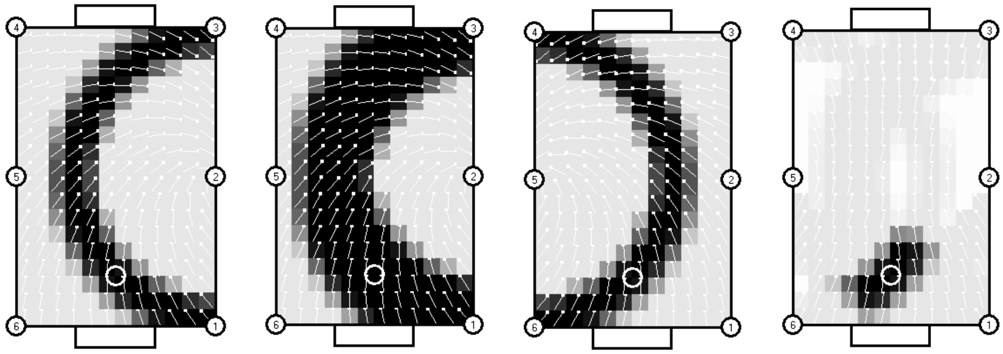


Figure 4: Intersection of angular information.



(a) (b) (c) (d)

Figure 5: Self-localization including angular information. The white arrows indicate the center of the directional fuzzy set at each cell.

The dilation operator modifies the h value at each cell of the gridmap G using the fuzzy dilation operator (2) above, and increases the Δ values by a constant value. $\theta, alpha$ and b are not modified. If desired, egomotion information could be included by using an appropriate structuring element B, and by shifting θ.

Fig. 5 repeats the same example as Fig. 3, again using real robot's data, but now we also use angular information. The orientation θ of the fuzzy set at each cell is shown by a white arrow. The result in (d) gives an estimate for both the position *and* robot's orientation. Note that the residual ambiguity in the previous example has been resolved.

Our self-localization technique has nice computational properties. Updating, blurring, and computing the center of gravity (CoG) of the fuzzy grid are all linear in the number of cells. In the Robocup domain we use a grid of 20×30 cells, corresponding to a resolution of 10 cm (angular resolution is unlimited since the angle is not discretized). Update, blur, and CoG all take less than 1 msec on a Pentium 400 MHz. On a 200×300 grid, corresponding to an environment of $20\,m \times 30\,m$, update, blur, and Cog respectively take 50, 100, and 17 msec.

6 Recovering from errors

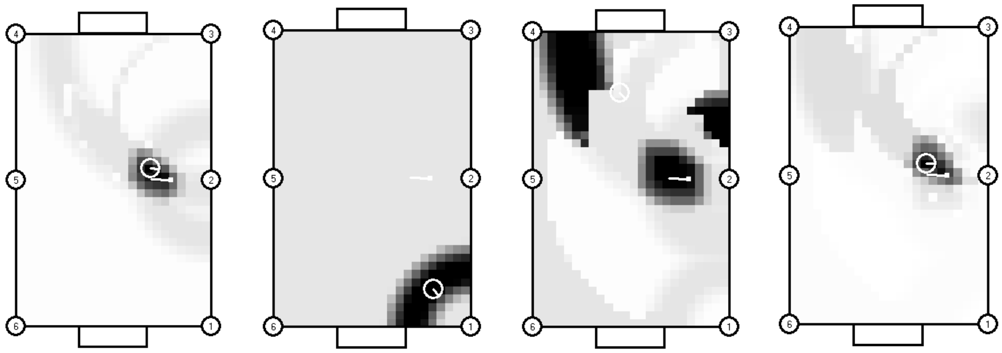
Finally, we show how the proposed technique can recover from dramatic localization errors produced by a mislabeling of a landmark. In the experiment shown in Fig. 6, the robot was started from a situation of total ignorance about its location, and put at the position marked by the white arrow. After having observed landmarks 2 and 3, the robot's possibility distribution G_t was the one shown in Fig. 6 (a) (the white circle marks the center of gravity). Unfortunately, in the next observation the robot mislabeled landmark 2 as landmark 1, thus producing the S_{t+1} shown in (b). Combining this S_{t+1} into the grid led to the catastrophic result shown in (c). Two further observations (landmark 2 and 3), however, were enough to bring this estimate back to the correct value (d).

It is interesting to note that what made the error dramatic in this case was the fact that the mislabeled landmark was only the second observation after having started from scratch. A similar error would only have a minor impact on the localization of the robot if it occurs after a longer history of observations.

7 Conclusions

The fuzzy position grid (FPG) provides a viable solution to the problem of localization of a legged robot in the Robocup domain using limited computational resources. This technique has shown to adequately address the challenges presented by this application, where motion estimates are highly unreliable, observations are uncertain, accurate sensor models are not available, and real time operation is of essence.

The only other localization algorithm reported to have been applied to the same domain is, to the bes of our knowledge, the SRL algorithm [8]. SRL extends MonteCarlo localization (MCL) by adding heuristics to quickly forget the past history when receiving observations that conflict with the current estimate. A drawback of SRL with respect to our fuzzy position grid is its high computational cost, compared with the negligeable complexity of the FPG. (In fact, computa-



(a) (b) (c) (d)

Figure 6: Recovery from a large self-localization error. The white arrow indicates the position of the real robot.

tion limitations forced the designers of SRL to ignore 50% of the observations.) Also, the forgetting heuristics of SRL may lead to a reduced robustness in the case of a landmark misidentification, and event which is correctly handled by the FPG.

The version of FPG presented here programmatically ignores odometric information. This version can still produce a new correct estimate every few seconds, i.e., every time the robot is pointing its camera to some landmark. In order to provide a faster update rate, we are now integrating some qualitative egomotion information into FPG along the lines suggested above. Other current work includes the systematic evaluation of our technique, and comparison with other ones, using an external motion capture system to collect data in real game situations. We also plan to test our technique on other robot platforms and environments, and using natural landmarks.

Acknowledgements The Swedish KK foundation provided support for this research. The motion capture system was kindly provided by Qualisys AB. The authors thank TeamSweden (www.aass.oru.se/Living/RoboCup/) for fruitful discussions.

References

[1] I. Bloch. Information combination operators for data fusion: A comparative review with classification. *IEEE Trans. on Systems, Man, and Cybernetics*, A-26(1):52–67, 1996.

[2] I. Bloch and H. Maître. Fuzzy mathematical morphologies: a comparative study. *Pattern Recognition*, 28(9):1341–1387, 1995.

[3] W. Burgard, D. Fox, D. Hennig, and T. Schmidt. Estimating the absolute position of a mobile robot using position probability grids. In *Proc. of AAAI*, pages 896–901, 1996.

[4] Sony Corporation. Entertainement robot AIBO. http://www.world.sony.com/robot/.

[5] The RoboCup Federation. Robocup official web site. http://www.robocup.org/.

[6] D. Fox, W. Burgard, and S. Thrun. Markov localization for mobile robots in dynamic environments. *J. of Artificial Intelligence Research*, 11:896–901, 1999.

[7] A. Gelb, editor. *Applied Optimal Estimation*. MIT Press, Cambridge, MA, 1974.

[8] S. Lenser and M. Veloso. Sensor resetting localization for poorly modelled mobile robots. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation*, San Francisco, CA, 2000.

[9] A. Saffiotti. The uses of fuzzy logic for autonomous robot navigation. *Soft Computing*, 1(4):180–197, 1997. http://aass.oru.se/~asaffio/.

[10] A. Saffiotti and L.P. Wesley. Perception-based self-localization using fuzzy locations. In M. van Lambalgen L. Dorst and F. Voorbraak, editors, *Reasoning with Uncertainty in Robotics*, pages 368–385. LNAI 1093, Springer, 1996.

[11] L. A. Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 1:3–28, 1978.