Let Robots Play Soccer under More Natural Conditions: Experience-Based Collaborative Localization in Four-Legged League

Qining Wang, Yan Huang, Guangming Xie, and Long Wang

Intelligent Control Laboratory, College of Engineering,
Peking University, 100871, Beijing, China
qiningwang@pku.edu.cn
http://www.mech.pku.edu.cn/robot/fourleg/

Abstract. This paper presents an experience-based collaborative approach for a group of autonomous robots to localize in asymmetric, dynamic environments. To help robots play soccer under more natural conditions, we propose a Markov localization based hybrid method with integration of environment experience construction and dynamic reference object based multi-robot localization. By using this method, the robot can estimate and correct its position perception more accurately and effectively among a group of autonomous robots, taking the odometry error and other negative influence into consideration. Satisfactory results are obtained in the RoboCup Four-Legged League environment.

1 Introduction

On the move to real human soccer conditions, current localization approaches applied in RoboCup (eg. [1], [2]) seem not enough. In the human soccer, there are two aspects which may inspire the self localization of mobile robot systems. On the one hand, the features surrounding the soccer field may be exploited as the sensory information in probabilistic approaches. Inspired by the features, some systems applied image-retrieval approach in localization [5]. However, the computational cost is expensive. Besides, the requirement of building a huge database is not practical, especially in complex environments. On the other hand, collaboration among the robot team may help self localization. Previous research in localization has proven that the cooperation in self-localization among multiple robots has impressive performance in real robot systems (see [3] for overview). The limitation of such robot systems is that the robot needs to identify other one precisely. It is difficult to perform collaborative localization for robots dealing with situations where they can detect but not identify other robots.

Our work focused on applying image-retrieval approach and collaboration in self localization in RoboCup. In the following section, we describe the method for individual localization with experience. In section 3, we present how to use the sharing information to improve the Markov localization when the robot can not localize accurately by itself. In section 4, satisfactory results on localization through our approach is shown in experiments using Sony Aibo ERS-7 robots.

U. Visser et al. (Eds.): RoboCup 2007, LNAI 5001, pp. 353–360, 2008.

2 Individual Localization with Experience

Based on [1]-[2], in our approach, the current position of the robot is modelled as the density of a set of particles which are seen as the prediction of the location. Initially, at time t, each location l has a belief:

$$Bel_t(l) \leftarrow P(L_t^{(0)} = l) \tag{1}$$

To update the belief of robot possible location, at first, this approach uses the new odometry reading o_t:

$$Bel_t(l) \leftarrow \int P(l|o_t, l^-) Bel_t(l^-) dl^- \tag{2}$$

Considering the mobile robot with complex motions, let the geometric center of robot body as the location vector ϕ, which contains the x/y- global coordinates of the center point. Another vector θ is defined as the heading direction. Then every particle is updated by the motion model as follows when the robot moves:

$$\phi_t = \phi_{t-1} + \Delta_t \tag{3}$$

where Δ_t represents the displacement in x/y coordinates and heading direction.

To implement image retrieval system in Markov localization, we divide the sensory update into two parts: updating position probability by landmark perception and experience matching. If the robot recognizes landmarks well enough, landmark based sensor model will update the belief of position with the new landmark reading s_t:

$$Bel_t(\phi_t) \leftarrow \beta P(s_t|\phi_t) Bel_t(\phi_t) \tag{4}$$

where β is a normalizing constant. We set $N_1(t)$ which is the amount of lasting frames of having no landmark perception from t as a condition to activate the experience system. If $N_1(t)$ is great enough, the experience based sensor model will update the probability as follows:

$$Bel_t(\phi_t) \leftarrow \gamma P(e_t|\phi_t) Bel_t(\phi_t) \tag{5}$$

where e_t is the new reading experience with γ being the normalizing constant.

2.1 Experience Construction

The feature that is exploited from images with no landmark in the view, and represents the invariant character of images obtained at positions where collisions and other negative effects more likely occur is defined as *Experience*.

In our method, we divide one image which is obtained by the robot camera into six parts. First, image features including average color value $f_{i,j}$ and color variance d_i in the divided areas are calculated by the following equations:

$$f_{i,j} = \frac{\sum\limits_{x,y} M[y][j][x]}{N_i}; \{j = 0, 2, 3; i = 1, 2, 3, 4, 5, 6\} \tag{6}$$

where $f_{i,j}$ is the average value in the color channel j of area i. $M[y][j][x]$ represents the value in the color channel j at the position (x, y) in the image. N_i is the number of the pixels in area i. Clearly, the $f_{i,j}$ is in the range from 0 to 255.

$$d_i = \frac{\sum\limits_{x,y}(|M[y][0][x] - f_{i,0}| + |M[y][1][x] - f_{i,1}| + |M[y][2][x] - f_{i,2}|)}{N_i} \tag{7}$$

where i=1, 2, 3, 4, 5, 6. d_i is in the range from 0 to 382.5. When the value of color variance in the certain area gets maximum, d_i is 382.5.

After calculating features in divided areas, we collect average color value F_j and color variance D in the whole image which are calculated by the following equations:

$$F_j = \frac{\sum_i f_{i,j}}{S}; \{j = 0, 1, 2\} \tag{8}$$

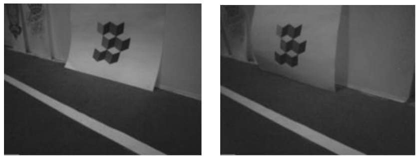
where F_j represents the average value in the color channel j of the whole image. S is the number of divided areas in the image.

$$D = \frac{\sum\limits_{i}(|f_{i,0} - F_0| + |f_{i,1} - F_1| + |f_{i,2} - F_2|)}{S} \tag{9}$$

where D is in the range from 0 to 382.5.

In our system, the invariant features of images include $f_{i,j}$, d_i, F_j, and D. All the features are calculated from images collected in certain places where the robot needs experience to help. We construct experience database embedded in robot's memory. This database stores the features along with the global coordinates of the position where the image is taken. All the features are calculated off-line and stored in the database as experience. When the experience module is activated, the feature of current image taken by camera is computed on-line notated as *imageFeature*. Meanwhile, the record notated as *bestRecord* whose feature is most similar to *imageFeature* is selected from the database. Fig. 1 shows the result of finding the best pose in database based on experience. The query image is on the left while its most similar image in the database is on the right. Their poses are represented by (x, y, θ). x, y are calculated in millimeter, while θ is in degree.

When the experience module is activated, difference between *imageFeature* and the feature of *bestRecord* is calculated. If the difference is small enough, the pose of *bestRecord* is transferred into *bestPose* notated as l_{best} which is in the form of world coordinates in the robot system. With such *bestPose*, probabilities of all the sample poses are updated and new pose templates which are random poses near the *bestPose* are generated to perform the resample procedure in Markov localization. It is true that the more experience in database, the more precisely the calculation is. However, building such database is expensive in time cost and even unreachable in complex environments. As a part of the sensor update module, experience can help the Markov localization converge as soon as possible, which means the robot can know own position immediately. In our approach, we only need to construct the database in those really difficult situations. This method works well in real robot applications.



(a) (b)

Fig. 1. Examples for finding the best pose in image database. Images in the database are collected in the areas of the field where the robot can not see any landmark every $100mm$ in x, $100mm$ in y and $45°$ in θ. (a) is the current image taken by robot's camera when its real position is $(-1660, 1520, 135°)$. (b) is the most similar picture to image (a) in the experience database which the corresponding position of the robot is $(-1600, 1500, 135°)$. The location error is $60mm$ in x, $20mm$ in y, and $0°$ in θ.

2.2 Self Learning in Experience Collection

One of the difficulties in applying image-retrieval system into real robot localization is how to collect the experience efficiently and correctly. We create a self learning method for experience collection. The robot can collect images along with corresponding positions autonomously. When construct the experience database, we use the black-white stripes to adjust robot body which is similar to the one used in gait optimization mentioned in [4]. In the self learning procedure, at first, the robot adjusts its own body to the initial position which is preset by our control system. By using the stripes, the robot walks to the next position and stops to capture images in left and right view respectively as shown in Fig. 2. The black-white stripes help robot go to the preset position precisely.

(a) (b) (c)

Fig. 2. Self learning procedure in experience collection. (a) shows the Black-white stripes for body adjusting. The robot captures image in the left view and right view as shown in (b) and (c) respectively.

3 Collaborative Localization

In RoboCup, static reference objects like beacon, and goal can be used to help localize in complex environments. However, global coordinates of such objects need to be known beforehand. Those static reference objects are not applicable in an unknown environment. To solve this problem, we propose the concept of *Dynamic Reference Object*. The object that can be detected by more than one robots among the team will be the candidate dynamic reference object. If the frequency of clearly recognizing the object is high enough, it may be set as the

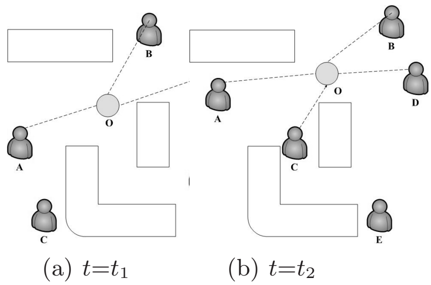


Fig. 3. A simple system with five mobile robots and a dynamic reference object: (a) At time t_1, robot A, B and E can see the dynamic reference object O. If at this time robot A, for example, needs the reference object to help, A will use the calculated position of the object from B or E. Querying the most possible position in team message shown in Table 1, A will take the calculated result by B as the reference. (b) At time t_2, C and D have not detected any landmark or experience for a period. Thus their answers to the object position is relatively unreliable. Position possibilities of them are shown to be low in Table 1. The reference object position will be set as B percepts.

dynamic reference object. There is no need to know the object's position as a precondition. If a robot can localize itself accurately, the position of the dynamic reference object calculated by this robot is reliable. Meanwhile, another robot that has seen the reference object can use this calculated position of the object to measure own location. This information is useful for decreasing the time cost of Markov localization convergence and improve the result of position estimate especially for multiple robots collaboration.

With the assumption that robots can communicate with each other, our approach integrates *Reference Object Position Possibility* in the team message which will be broadcasted to every robot. The item which is relevant to the object position in team message includes calculated position, robot ID, time, and position possibility. This position possibility is due to the accuracy of the robot self localization. In our system, the object position possibility is notated as P_r is measured by the following equation:

$$P_r = P_l e^{-\mu^2} + P_e e^{-\omega^2} \tag{10}$$

where P_l and P_e are certain probabilities for landmark and experience update respectively. μ is the sum of lasting frames after detecting the latest landmark, while ω is the sum of lasting frames after exploiting good experience. In real robot application, P_r will be normalized less than 1. If P_r is high enough, the calculated result by this robot will be the most reliable one among different robots perception. A robot that needs help always uses the most possible position of the reference object at the same time when it detects the object by itself. To illustrate the method, a common robot system is shown in Fig. 3 with five mobile robots. Object O is supposed to be the dynamic reference object. Table 1 is the real-time information in team message of the system in Fig. 3.

Table 1. Team message relevant to dynamic reference object

Calculated Position	Robot ID	Time	Position Possibility
$(2388, 700)$	A	t_1	0.71
$(2264, 658)$	B	t_1	0.92
$(2530, 710)$	E	t_1	0.86
$(2368, 803)$	A	t_2	0.81
$(2401, 801)$	B	t_2	0.91
$(2103, 743)$	C	t_2	0.32
$(2215, 725)$	D	t_2	0.43

In our approach, collaboration is a part of probability update modules in Markov localization. There is a problem that robots should known when to activate the collaboration module using the dynamic object as a reference. To improve Markov localization using our collaborative approach, the collaboration module will be activated in two situations. We set $N_2(t)$ by using as the sum of lasting frames of having no landmark perception or experience as a condition to activate the collaboration system. If $N_2(t)$ is great enough and the robot has detected the dynamic reference object, the collaboration module will update the probability of every poses. In addition, if the robot has a perception of the object which has a relatively high position possibility, the robot will use this reference to improve the Markov localization in a collaborative way.

4 Experimental Results

The experience-based collaborative approach presented above has been implemented on the Sony Aibo ERS7 legged robot in RoboCup environment. Fig. 4(a) shows the environment in 2007. In our localization experiment field, we use the field similar to the standard field in four-legged soccer field 2007. However, we remove the beacons. As shown in Fig. 4(b), our field is surrounded by colorful advertisement which simulates the real human soccer environment.

(a) (b) (c)

Fig. 4. Experimental field. (a) is the soccer field with two colorful beacons in 2007. (b) shows field with no beacon which is used to test our localization approach. (c) is the colorful advertisement placed around our test field.

4.1 Individual Robot Localization

We randomly select 8 points to test the self localization results. The robot is expected to go to the preset positions through localization. When it stops, we

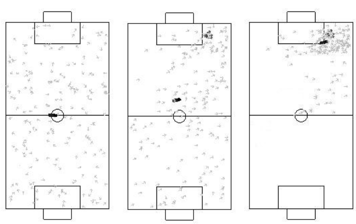
Table 2. Results of self localization in randomly walking

Point Number	Expected Position (x, y, θ)	Real Position (x, y, θ)	Error (x, y, θ)
1	$(-1290, -440, 15)$	$(-1496, -713, 147)$	$(206, 273, 132)$
2	$(-1450, -300, 0)$	$(-1410, -150, 0)$	$(40, 150, 0)$
3	$(-180, -670, 45)$	$(-230, -610, 9)$	$(50, 60, 36)$
4	$(1430, -250, 55)$	$(-1909, -1162, 132)$	$(461, 912, 76)$
5	$(-650, 170, 0)$	$(-404, -427, 5)$	$(246, 597, 5)$
6	$(270, -480, -90)$	$(102, -402, -48)$	$(168, 78, 42)$
7	$(-1440, -340, 10)$	$(-1322, -332, 5)$	$(78, 8, 5)$
8	$(-2160, -390, 0)$	$(1979, -454, 8)$	$(181, 64, 8)$

calculate the real positions on the ground. Table 2 shows the results in detail. x, y are calculated in millimeter, while θ is in degree.

4.2 Collaborative Localization

In this experiment, the orange ball used in the four-legged league is considered as the *dynamic reference object*. We use three robots to perform multi-robot localization. Every robot uses the hybrid system tested in the individual experiment mentioned above. We set one of the three robots as a sample to estimate our collaborative approach. The other two robots move randomly to catch the ball and broadcast the ball position with position possibilities mentioned in section 3. We receive the calculated result from the sample robot. Only experience and collaboration can help the robot localize. The localization result of the sample robot which has used the collaborative approach is shown in Fig. 5. The probability distribution can converges quickly after 3-9 seconds when the dynamic reference object is taken into account.



(a) $t = 0s$ (b) $t = 3s$ (c) $t = 9s$

Fig. 5. The localization result of applying collaborative approach with dynamic reference object. Solid arrows indicate MCL particles(100). The calculated robot position is indicated by the solid symbol. (a) is the initial uniform distribution. (b) is the calculated result after 3 seconds. (c) is the well localization result after 9 seconds.

5 Conclusion

In this paper, we have demonstrated an experience-based collaborative approach that combines image database for experience without landmarks and real-time sensor data for vision-based mobile robots to estimate their positions under more natural conditions towards real human soccer environment. On the one hand, our approach presented a fast and feasible system for vision-based mobile robots to localize in the dynamic environment even if there is no artificial landmark to help. On the other hand, we showed the collaborative method with introduction of Dynamic Reference Object to improve the accuracy and robustness of self localization, even in the circumstance that the robot can not localize individually or has no idea of who is nearby.

Acknowledgments

This work was supported by the 863 Program of China (No. 2006AA04Z258) and 985 Project of Peking University.

References

1. Fox, D., Burgard, W., Dellaert, F., Thrun, S.: Monte Carlo localization: Efficient position estimation for mobile robots. In: Proc. of the National Conf. on Artificial Intelligence, pp. 343–349 (1999)
2. Röfer, T., Jüngel, M.: Vision-Based Fast and Reactive Monte-Carlo Localization. In: Proc. of the IEEE ICRA, pp. 856–861 (2003)
3. Arkin, R.C., Balch, T.: Cooperative multiagent robotic systems. In: Kortenkamp, D., Bonasso, R.P., Murphy, R. (eds.) Artificial Intelligence and Mobile Robots. MIT/AAAI Press, Cambridge (1998)
4. Röfer, T., et al.: GermanTeam RoboCup 2004, tech. rep. (2004), `http://www.germanteam.org/GT2004.pdf`
5. Wolf, J., Burgard, W., Burkhardt, H.: Robust vision-based localization by combining an image-retrieval system with Monte Carlo localization. IEEE Trans. on Robotics 21(2), 208–216 (2005)
6. Fox, D., Burgard, W., Kruppa, H., Thrun, S.: A probabilistic approach to collaborative multi-robot localization. In: Autonomous Robots, vol. 8(3), pp. 325–344 (2000)