A Method for Localization by Integration of Imprecise Vision and a Field Model

Kazunori TERADA[1], Kouji MOCHIZUKI[1], Atsushi UENO[1], Hideaki TAKEDA[1], Toyoaki NISHIDA[2], Takayuki NAKAMURA[1], Akihiro EBINA[1], and Hiromitsu FUJIWARA[1]

[1] Graduate School of Information Science
Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara 630-0101, Japan
{kazuno-t, kouji-m, ueno, takeda, takayuki, akihi-e, hiromi-f}@is.aist-nara.ac.jp

[2] Department of Information and Communication Engineering
School of Engineering
The University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
nishida@kc.t.u-tokyo.ac.jp

Abstract. In recent years, many researchers in AI and Robotics pay attention to RoboCup, because robotic soccer games needs various techniques in AI and Robotics, such as navigation, behavior generation, localization and environment recognition. Localization is one of the important issues for RoboCup. In this paper, we propose a method of robot's localization by integrating vision and modeling of the environment. The environment model that realizes the robotic soccer filed in the computer can produce an image of robot's view at any location. In the environment model, the system can search and appropriate location of which view image is similar to the view image by the real robot. Our robot can estimate location from goal's height and aspect ratio on the camera image. We search the most suitable position with hill-climbing algorithm from the estimated location. We programmed this method, and tested validity. The error range is reduced from 1m$\sim$50cm by robot's estimation from 40cm$\sim$20cm by this method. This method is superior to the other methods using dead reckoning or range sensor with map because it does not depend on the field size on precision, and does not need walls as landmark.

1 Introduction

In the domain of the robotic soccer, there are various classes of problems; navigation, behavior generation, recognition of the environment, and localization. In these problems, localization is especially indispensable technique for robotic soccer. In order to generate cooperative behavior, an agent needs to know its position within the environment. Therefore, the problem of estimating the position of a mobile robot is one of the fundamental problems in the field of mobile

M. Veloso, E. Pagello, and H. Kitano (Eds.): RoboCup-99, LNAI 1856, pp. 509–518, 2000.

robots. In this paper we propose a method of localization by integration of imprecise vision and a 3D environmental model. We will show that this method is also effective to identify opponent robots.

Dead reckoning and range finder are often utilized to the localization problem [7] [5] [3]. Dead reckoning uses odometry (i.e. counting wheel rotations) to determine the robot's position. Since, errors from slip of wheels accumulate over time, estimation of its position becomes increasingly inaccurate. The range finders such as sonar and IR sensors are not useful in the environment in which there are no walls which reflect the ultrasonic wave and infrared rays. In order to overcome this problem we need to use the method of localization there are less dependent on its environment.

In this paper we propose the system which uses both visual sensor and the soccer field model. By making comparison between images from the visual sensor and provided by the field model, the robot can estimate its position.

In the domain of vision-guided mobile robot research, there are a lot of techniques of localization using visual sensors. There have been various methods which use 3D model of environments to estimate the robot's position and orientation. In this approach, 3D model is employed to generate an expected image and it is compared with an image captured by the robot [2] [9] [6] [4] [1].

If the 3D model of environment is already known and we can measure several feature points of an object in a captured 2D image, the distance to object may be derived by geometrical model. However, it is hard to measure such feature points because of the following two reasons. One reason is that if there is an unknown object in the captured image and it obstructs the view, these feature points would not be observed. The other is that the image processing system is often confronted with change of lighting, therefore accurate measurement is impossible. In order to overcome this problem, we add a method of matching two images to distance estimation from several feature points.

2 System Architecture

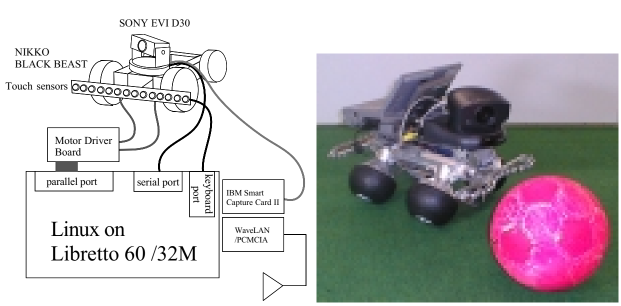

Fig. 1. Our soccer robot.

Our robot consists of four major hardware components; a portable PC (Libretto 50 or 60, 100, Toshiba), a vision system including a camera and a video capture card, a tactile sensor system, and a chassis including motors and a motor drive board. The detail of our system is described in [8].

An image captured by the vision system is processed by the vision module. The vision module provides information about objects in the field and the image is segmented into 7 colors (See Table1) regions in the vision module which runs on PC on the robot.

Table 1.

object	color
ground	green
one goal	yellow
another goal	blue
wall	white
ball	red
outside of field	black
robot	gray

Every pixel in an image is classified to 6 colors except gray by means of a discrimination rule which uses the Mahalanobis distance, and this rule is learned by sample color data. A pixel which does not classified to any colors labeled to gray.

3 localization

The procedure of our method of localization consists of the following two steps; 1) Estimate the current position by calculating distance to a landmark from an image which is sent from camera on the robot, 2) Revise the position by comparing the camera image and vision images that are generated by the field model.

We select a goal as a landmark to estimate the position. The reasons of selecting the goal as a landmark are as follows. 1) The robot can always see either goal everywhere in the soccer field. 2) The robot can always see a complete view of either goal since it can rotate its camera.

3.1 Estimation of the position

We use distance and angle from the goal estimate position of the robot in the field. Distance and angle are calculated from width and height of the goal that are identified in an image captured by the camera on the robot. The relations between distance, angle and width, height of the goal are as follows;

– The height of a goal in the image is inversely proportional to the real distance to the goal.
– The angle to the goal concerns to the ratio of the width to the height of goal.

The former relation can be easily formulated;

$$d = \frac{a}{h} \tag{1}$$

Where the h is height of the goal in the image. We select the constant a by calculating the mean value from the real data. The latter is derived from the geometrical model of the relation between a goal and a robot(See Figure 2).

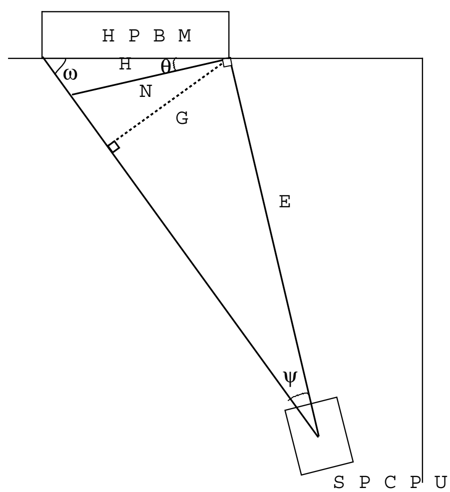

Fig. 2. The geometrical model of the relation between a goal and a robot

According to this model, angle to the goal is;

$$\theta = 90 - \arcsin(\frac{d}{g}\sin(\arctan(\frac{m}{d}))) - \arctan(\frac{m}{d}) \tag{2}$$

We define the origin of the coordinate axes as center of center circle in the soccer field. The position of robot is;

$$\begin{cases} x = 4110 - d\cos(\theta) \\ y = \qquad d\sin(\theta) \end{cases}$$

The position calculated this method has about 60cm error in average. This error is caused by limit of image precision (64 × 48 pixels) and error of color segmentation. Image precision is limited because all kinds of processing including image processing are done in a single computer in our robot and it requires real-time response.

In order to reduce this error, we revise the position with the field model.

3.2 Position Revision

The position revision system consists of a server which revises the position, and clients each of which is on a robot and send estimated positions to the server. The server communicates with multiple clients and holds the position of each robot. Figure 3 shows the client-server system applied in our work.

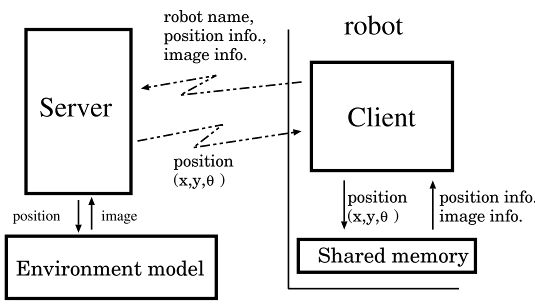

Fig. 3. Server Client System

The role of the server and the client is described as follows. Server receives both data of the position which is estimated in the computer on the robot, and images captured by the camera on the robot through wireless ethernet. The server then revises the position by comparing the two images; one is captured by the robot, another is generated in the server by means of a 3D field model.

3.3 Field Model

The server has a 3D field model. The features of this model are;

1. The size of every part of field such as wall and goal can be changed.
2. This model can generate any image of view that the robot ought to see from coordinates (x, y) and angle θ.
3. We can place multiple robots in the field.
4. The shape of every robot can be changed.

3.4 Method of Position Revision

We use two types of images for position revision. One type of images that is captured by the camera on a robot. The image is segmented to 7 colors region by the vision module on the robot and sent to the server through ethernet. The other type of images is an ideal image of the field. This image is generated by the server and is calculated from value of position; (x, y, θ) that is sent from a client to the server. The strategy of position revision is to find an optimal point in the field model. In this point the field model generates the most similar image to the captured image. The method of finding this point is to search the neighborhood point of starting point which is estimated from geometrical model. We use the hill-climbing algorithm as a search algorithm.

The evaluation function is;

$$func(a, b, c) = \frac{2c}{a + b} \tag{3}$$

a is the number of pixels of the goal area in an image which generated by the 3D field model. b is the number of pixels of the goal area in an image which is captured by camera on the robot. c is a number of intersection pixels of both areas when two images are overlapped in order to fit center of gravity of both areas.

By means of hill-climbing algorithm search of optimal point is carried out by the following steps.

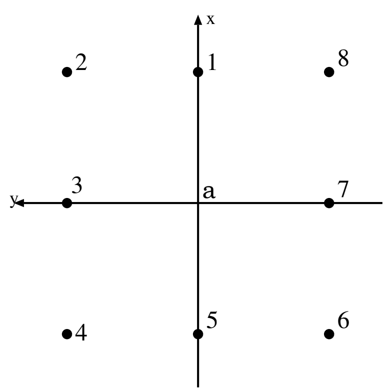

Fig. 4. Neighbor point of (i, j)

1. Step1: Evaluation of beginning of search.
   (a) Calculate the value of beginning point $p_t(i, j)$ by evaluation function.
2. Step2: Calculate the value of the next candidate points
   (a) Select the neighbor 8 points of $p_t(i, j)$ (See Figure 4).
   (b) Generate the image of each point and calculate a value by the evaluation function.
   (c) Select the point $p_{t+1}(i, j)$ in which the evaluation value is the greatest of 8 points.
3. Step3: Comparison.
   (a) If the value in $p_t(i, j)$ is greater than that in $p_{t+1}(i, j)$, finish searching.
   (b) If the value in $p_t(i, j)$ is greater than 0.99, finish searching.
   (c) If the value in $p_{t+1}(i, j)$ is the same as that in $p_{t-1}(i, j)$, finish searching.
   (d) If the value in $p_{t+1}(i, j)$ is greater than that in $p_t(i, j)$, return Step2.

4 Experimental result

We had experiments on the proposed method. We put the robot on the point of the field that is randomly selected, and executed the program. Experimental results are shown in Table 2.

Table 2. Comparison of errors

position	estimated	revised
(2000.0)	659mm	206mm
(1850,0)	639mm	308mm
(1300,0)	1113mm	320mm
(1150,0)	1111mm	381mm
(1000,0)	927mm	335mm
(1000,1000)	279mm	260mm
(0,1000)	486mm	172mm
(-1000,1000)	606mm	243mm
(1000,-1000)	341mm	141mm
(0,-1000)	258mm	215mm
(-1000,-1000)	404mm	295mm

The mean value of the error before revision is about 600mm and it is reduced to 260mm after revision. While the error before revision becomes larger with the robot goes away from the goal, the error after revision becomes less sensitive to the distance. This fact suggests that our proposed method is tolerant of the changing position in the field. But, the error after revision becomes also smaller when the robot is closer to the goal. The reason for this is that we use the number of pixel in an image as a parameter of evaluation function. Since error rate does not depend on the amount of these pixels, the larger amount of these pixels contributes to make position more accurate.

5 Identification of Opponent Robots

In this section we explain a method of identification of opponent robots. This method is realized by our proposed localization method and field model. To distinguish team mate robot from opponent robot is important ability for soccer robot. An innocent robot may pass the ball to an opponent robot. Without explicit mark, to do that is considerably hard.

Our basic idea is that there are only team mate robots in the field model constructed in the server. The server know all positions of each team mate robot by communicating with each robot, and can generate an image in which there are only team mate robots. The procedure of distinguishing a team mate from opponent robots is follows.

1. Preparation of two images; one is generated by the server, and another is sent from one of the robot in the field.
2. Counting the pixels which are colored gray. Each image is segmented to 7 colors and the robot is colored gray in each image.
3. Calculation of remainder of each number of pixels; r.
4. If r is greater than a threshold, we define that there is at least an opponent robot in the image which is captured by robot. We select the threshold as 100 pixels.

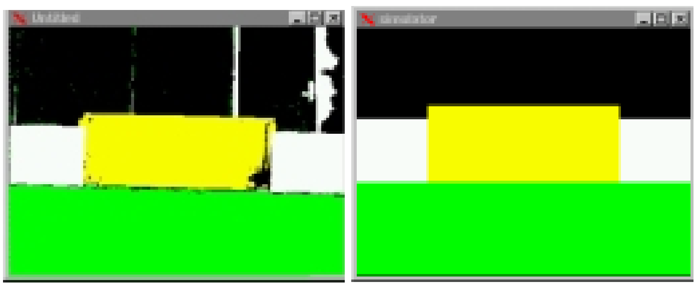

Fig. 5. There are no robots.

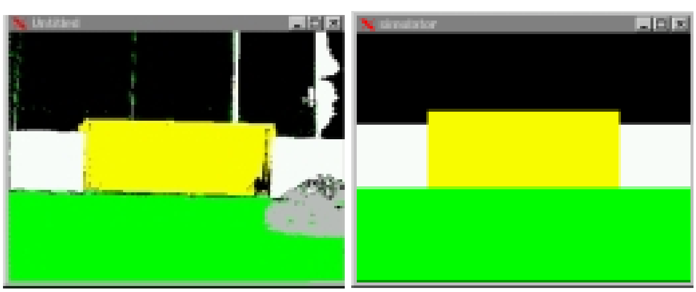

Fig. 6. There is only opponent robot.

4 experimental result are shown in Figure 5 $\sim$ Figure 8. In each figure, (a) is captured by a camera on a robot and (b) is generated by server. In figure (b) a team mate robot is expressed by a gray cone.

In Figure 5(a), the number of gray pixels is 66 and in Figure 5(b) 0. Therefore, the server decides that there are no robots. In Figure 6(a), the number of gray pixel is 184 and in Figure 6(b) 0. Therefore, the server decides that there is a team mate robot. In Figure 7(a), there is a robot in the left side of the goal and

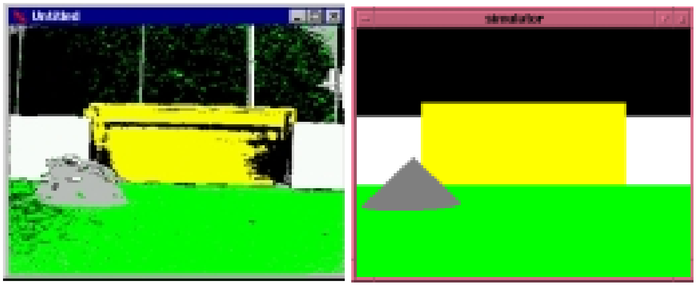

Fig. 7. There is only a team mate robot.

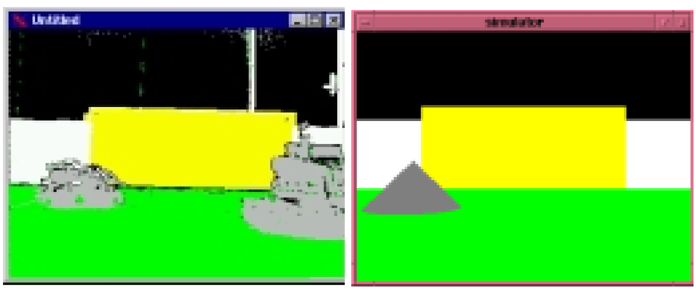

Fig. 8. There are both a team mate and an opponent robot.

the number of gray pixel is 155. In Figure 5(b) there is a corn as a team mate robot. The server know each position of each team mate robot, and can generate an image in which there is a team mate robot. The number of gray pixel is 104. Therefore the server decides that there is a team mate robot. While in Figure 8(a), there are two robots, in Figure 8(b), there is one robot, the server decides that there are both a team mate and an opponent robot.

6 Conclusions

In this paper, we propose a method of robot's localization by integrating vision and modeling of the environment. We also propose a method of identification of opponent robots by using vision and the field model. By means of our method, accuracy of the localization of the robot in the soccer field is improved.

In the method of using geometrical model and image's feature points, a noise of image prevents sampling of the feature value and directly effects the accuracy of the localization. On the other hand, in the method of comparing an image captured by robot and an image generated by the field model, starting point of search is important to reduce the computational cost. Combining these two methods, we reduced the cost of search and improved the accuracy of localization.

There are two advantages of our localization method against the other method such as a method utilized sonar or IR sensor and a method of dead reckoning. One is that our method does not depend on the size of the field. We utilize the view of goal in order to estimate and revise the position of the robot, therefore if only the goal is visible, localization is possible. If the size of the field become greater, errors from dead reckoning are proportionally increased. The other is that our method does not depend on the structure of the field. In the method utilized sonar or IR sensor, if there are no walls, the robot could not know its position.

In the method of identification of the opponent robot, we compare an image of a field model in which there are only a team mate robot to an image captured by the robot in order to distinguishing a team mate robot from opponent robots. Our experimental result shows that we can identify the opponent robot without explicit marking.

In our approach a problem still exists; If the robot can not see landmark, the estimation of position by matching two images is impossible. In order to overcome this problem we can use multiple landmarks and multiple images captured by a camera while the camera is rotating.

References

[1] H. Christensen, N. Kirkeby, and et al. Model-driven vison for in-door navigation. *Robotics and Autonomous Systems*, 12:199–207, 1994.

[2] C. Fennema, A. Hansen, E. Riseman, J.R. Beveridge, and R. Kumar. Model-directed mobile robot navigation. *IEEE Trans. on Systems, Man, and Cybernetics*, 20:1352–1369, 1990.

[3] J.-S. Gutmann, W. Hatzack, I. Herrmann, B. Nebel, F. Rittinger, A. Topor, T. Weigel, and B. Welsch. The cs freiburg robotic soccer team: Reliable self-localization, multirobot sensor integration, and basic soccer skills. In *RoboCup-98: Robot Soccer World Cup II*. Springer-Verlag, Berlin, Heidelberg, 1998.

[4] A. C. Kak, K. M. Andress, and et al. *Hierarchical evidence accumulation in the PSEIKE system and experiment in model-driven mobile robot navigation.* M. Henrion and R. Shachter and et al., Elsevier, 1990.

[5] L. Kleeman. Optimal estimation of position and heading for mobile robots using ultrasonic beacons and dead-reckoning. In *Proceedings 1992, IEEE International Conference on Robotics and Automation*, pages 2582–2587, 1992.

[6] A. Kosaka and A. C. Kak. Fast vision-guided mobile robot navigation using model-based reasoning and prediction of uncertainties. *Computer Vision, Graphics and Image Processing -IU*, 56:271–329, 1992.

[7] E. Krotkov, L. Chrisman, F. Cozman, and G. Heredia. Dead reckoning and visual landmarks for a prototype lunar rover. In *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems 1995 (IROS95)*, 1995.

[8] T. Nakamura, K. Terada, and et al. The robocup-naist: A cheap multisensor-based moblie robot with on-line visual learning capibility. In *Proceedings of the second RoboCup Workshop.*, pages 291–303, 1998.

[9] T. Tsubouchi and S. Yuta. Map-assisted vision system of mobile robots for rechoning in a building invironment. In *Proceedings of 1987 IEEE International Conference on Robotics and Automation*, pages 1978–1984, 1987.