Team description of Team ARAIBO

Tamio Arai[1], Ryuichi Ueda[1], Shogo Kamiya[2], Toshifumi Kikuchi[2], Kohei Sakamoto[1], Yoshiaki Jitsukawa[1], Masaki Komura[2], Hisashi Osumi[2], and Kazunori Umeda[2]

[1]Department of Precision Engineering, School of Engineering, The University of Tokyo

[2]Department of Precision Mechanics, School of Science and Engineering, Chuo University

sonyrobocup@prince.pe.u-tokyo.ac.jp

http://www.arai.pe.u-tokyo.ac.jp/RoboCup

1 Introduction

Team ARAIBO (Advanced Robotics with Artificial Intelligence for Ball Operation) has taken part in RoboCup since 1999. Even though we could not get the top three positions in Four-legged Robot League, we have proposed several novel issues such as jumping heading, uniform Monte Carlo localization, dynamic programming, memory squeeze by strategy maps and so forth. We always tried to introduce theoretical approach into AIBO's maneuver.

The team consists of two universities: The University of Tokyo and Chuo University. It is hard to continue this kind of research because RoboCup has not been supported by industry. We always get the budget for student travel expense but it is hard to get enough budgets even for three students. Even though many universities sent more than ten students, it is just a dream for our team ARAIBO. However, we try to participate at least three more years, that is, totally 7 times from 1999 to 2005.

We focused onto two fundamental characteristics of the robots:

(A) Precise and prompt localization
(B) Robust and appropriate strategy of individual soccer player

To improve the ability of AIBOs, we have studied the three research themes:

(1) Representation of information uncertainty of states [1],
(2) Utilization of the uncertainty representation [2, 3],
(3) Enhancement of maneuvers: walking [4], attractive ball operations [5], and to accelerate the research we have developed
(4) A graphic simulator with camera models.

We discuss them in this report.

2 Representation and Utilization of Information Uncertainty

In RoboCup soccer, each robot determines its position and orientation, which is called "pose", and then decides a motion according to the situation of the field. However it is not always possible to measure the pose because of errors on sensors. Thus self-localization is still a key issue. In the early stage of RoboCup, errors generated in the process of sensing are recognized inevitable and thus neglected. But for these three years, we evaluated the error quantitatively and utilize the uncertainty of the sensing devices.

2.1 Self-localization

It is common knowledge in RoboCuppers that Bayes' theorem is one of the most effective methods for self-localization. Localization methods introduce estimation results with a probability density function (PDF), which returns a probability density of a robot's existence at a pose. In other words, a robot always evaluates the correctness of the estimated pose. Bayes' theorem assumes that sequence of sensor readings generates based on a PDF. We utilize image processing as the sensor readings, which has a great error because of disturbance of color detection.

The probabilistic method is powerful. However, neighboring results of image processing in the sequence are sometimes closely dependent each other. In this case, the PDF converges wrongly. We have solved this problem with the use of uniform distribution [1]. Convergence of PDF never occurs if Bayes' theorem is applied only with uniform distribution. On the contrary, PDF sometimes becomes zero all over the domain after a wrong result of image processing or a transfer of the robot by a judge. If the PDF becomes the zero state, we restore the PDF before the zero state and expand it.

2.2 Navigation with uncertain self-localization results

Next, we should utilize the uncertainty representation for action of the robot. We have studied this theme for several years [2, 3]. These studies use dynamic programming (DP) [6] for path planning. As the result of DP, a state-action map is computed. Since the state is represented as more than 7 degrees of freedom, the size of the map turns as large as 500 MB, and therefore too large for the RAM on AIBO's computer. Thus, high ratio compression of the output is essential [3, 7].

In this year, we attempt to use the PDF to calculate a proper walking direction [8]. We implement state-value function, which is output of dynamic programming, on a robot. An expected state value can be calculated from this state-value function and the PDF. The robot moves as the expected state value is maximized. When the task defined in dynamic programming phase does not need accurate self-localization result, the expected state value increases even if the PDF is ambiguous.

3 High speed walking

Enhancement of walking speed is a top priority of this year, since it is still the most important issue to win a game in the league. As shown in Fig. 1, we have succeeded in the enhancement of the walking speed. We have found that manipulability is a key for the speedup. When the robot is standing without moving, knees of the rear legs are bent as they have the maximum manipulability. The robot can move in any direction with high speed from this standing pose.

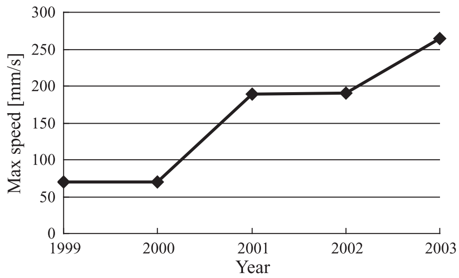

Fig. 1. Past and current walking speed of ARAIBO

4 Simulator for Image processing

There are various simulators in the league. We also develop a novel simulator [9]. The graphical interface of this simulator is shown in Fig.2. The simulator has client/server structure. A server make a virtual fields with several players in the computer and sends current states of everything on the field, i.e. poses of all the robots and the ball. Each player is controlled by a client, an agent. The client obtains the state from the server, and then it constructs a virtual camera image with Open-GL, and determines its action, then the action is sent to the server. The program of the client has the same code of a real robot. This part does image processing, self-localization, and decision making as a real robot.

The uniqueness of this simulator is the reality of the virtual camera images. We analyze the CMOS camera of ERS-210, and determine some types of noises and distortions. The virtual images are reflected the noises and distortions. Shake of the camera is also modeled. Therefore, we can develop image processing algorithms in the simulated environment.

5 Conclusion

We introduced some current studies of team ARAIBO with some references that are useful for understanding each study precisely. They include recognition, decision making, physical of the robot, and simulator with analysis of the CMOS camera.

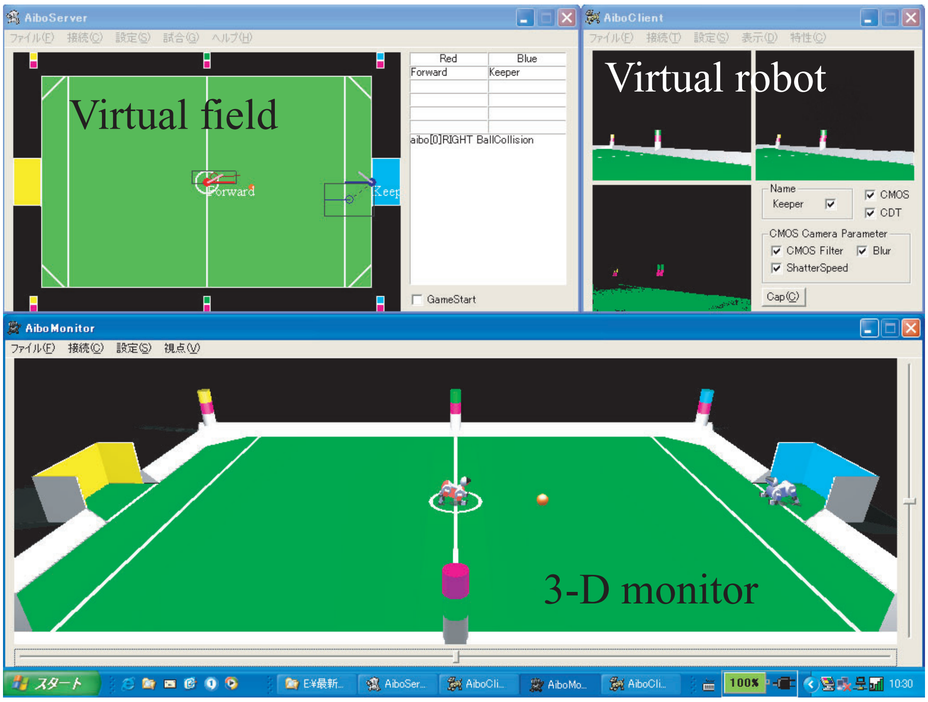

Fig. 2. Simulator

References

1. R. Ueda *et al.*, "Uniform Monte Carlo Localization – Fast and Robust Self-localization Method for Mobile Robots," in *Proc. of IEEE ICRA*, pp. 1353–1358, 2002.
2. T. Fukase *et al.*, "Quadruped Robot Navigation Considering the Observation Cost," in *A. Birk, S. Coradeschi, and S.Tadokoro (Eds.): RoboCup 2001: Robot Soccer World Cup V*, pp. 350–355, 2001.
3. T. Fukase *et al.*, "Real-time Decision Making under Uncertainty of Self-Localization Results," in *Proc. of 2002 International RoboCup Symposium*, pp. 327–379, 2002.
4. R. Ueda *et al.*, "Team description of Team ARAIBO," in *Proc. of 2002 International RoboCup Symposium (CD-ROM)*, 2002.
5. Y. Kobayashi and H. Yuasa, "Team ARAIBO," in *RoboCup1999*, pp. 758–761, 1999.
6. R. Bellman, *Dynamic Programming.* Princeton University Press, 1957.
7. R. Ueda *et al.*, "Vector Quantization for State-Action Map Compression," in *Proc. of ICRA-2003*, taipei, taiwan, 2003.
8. R. Ueda *et al.*, "Mobile Robot Navigation based on Expected State Value under Uncertainty of Self-localization," in *Proc. of IROS*, 2003.
9. K. Asanuma *et al.*, "Development of a Simulator of Environment and Measurement for Autonomous Mobile Robots Considering Camera Characteristics," in *Proc. of RoboCup International Symposium*, 2003.