

CIT Brains KidSize Robot: RoboCup 2014 Best Humanoid Award Winner

Yasuo Hayashibara¹(✉), Hideaki Minakata¹, Kiyoshi Irie¹,
Taiki Fukuda¹, Victor Tee Sin Loong¹, Daiki Maekawa¹, Yusuke Ito¹,
Takamasa Akiyama¹, Taitiro Mashiko¹, Kohei Izumi¹,
Yohei Yamano¹, Masayuki Ando¹, Yu Kato¹, Ryu Yamamoto¹,
Takanari Kida¹, Shinya Takemura¹, Yuhdai Suzuki¹,
Nung Duk Yun¹, Shigechika Miki², Yoshitaka Nishizaki³,
Kenji Kanemasu⁴, and Hajime Sakamoto⁵

¹ Chiba Institute of Technology, 2-17-1 Tsudanuma, Narashino, Chiba, Japan
yasuo.hayashibara@it-chiba.ac.jp

² Miki Seisakusyo Co., Ltd., 1-7-28 Ohno, Nishiyodogawa, Osaka, Japan

³ Nishizaki Co., Ltd., 1-7-27 Ohno, Nishiyodogawa, Osaka, Japan

⁴ Yosinori Industry, Ltd., 1-1-7 Fukumachi, Nishiyodogawa, Osaka, Japan

⁵ Hajime Research Institute, Ltd., 1-7-28 Ohno, Nishiyodogawa, Osaka, Japan
sakamoto@hajimerobot.co.jp

Abstract. In this paper, we describe the system design of the robots developed by our Team CIT Brains for the RoboCup soccer humanoid KidSize league. We have been participating in the Humanoid League for eight years. Two years ago, we redesigned the system to put a large weight on maintainability and usability. In RoboCup 2014, we won the first prizes of 4on4 soccer and technical challenge. Consequently, we were awarded the Louis Vuitton Humanoid Cup. The system we developed has high mobility, well-designed control system, position estimation by a monocular camera, user-friendly interface and a simulator. The robot can walk speedily and robustly. It also has a feedback system with a gyro sensor to prevent falls. It detects positions of landmarks by color-based image processing. A particle filter is employed to localize the robot in the soccer field fusing the motion model and landmark observation.

Keywords: Humanoid robots · Programming environment · Education robotics

1 Introduction

In this paper, we describe our system for the RoboCup soccer humanoid KidSize league. In RoboCup 2014 Brazil, we won the first prizes of 4on4 soccer and technical challenge. The results are indicated in Table 1. Consequently, we were awarded the Louis Vuitton Humanoid Cup. We started to develop the autonomous humanoid soccer robot in 2006 and have developed 22 robots in total. A history of the major KidSize robots in CIT

Brains is shown in Fig. 1. From 2013, we use the fourth generation robots. Through the development, we have studied many issues related to our robots [1–13].

CIT Brains is a joint team consisting of Hajime Research Institute and Chiba Institute of Technology (CIT). Hajime Research Institute developed mechanisms and prototypes of control systems of the robots [14]. CIT developed computer systems and overall intelligence such as perception and planning. CIT also made contributions to improve the mechanisms and control systems. We would like to emphasize that 13 members out of 19 members from CIT are undergraduate students. Any students who want to join this development can join our team. Senior students teach new members from the basic knowledge of the robot system. We aim to make an educational and research platform of intelligent humanoid.

Table 1. Results of CIT Brains in RoboCup2014

Soccer 4 on 4	8 wins - 0 loss Total goals: 33 goals - 3 losses
Technical challenge	High-Kick Challenge: 10 points Obstacle Avoidance and Dribbling: 10 points Artificial Grass Challenge: 7 points Total: 27 points

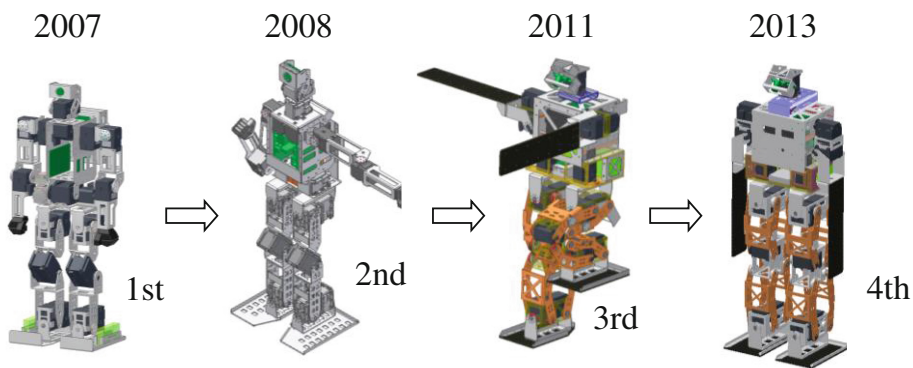


Fig. 1. A history of KidSize robots in CIT Brains

2 Overview of the System

A photograph of our robot is shown in Fig. 2. The specifications of the robot are summarized in Table 2. An overview of the system hardware is shown in Fig. 3. Our robot system consists of a USB camera, a computer board, an inertial measurement unit (IMU), 17 servomotors, a battery and several user interfaces such as switches.

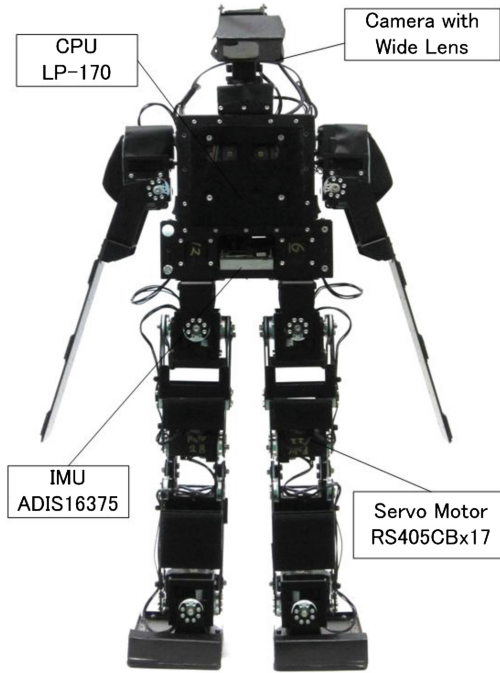


Fig. 2. Structure of the robot

Table 2. Specifications of the robot

Weight	3.5 kg (Including batteries)
Height	600 mm
Maximum velocity	0.4 m/s
Walking directions	All direction and rotation (select the angle, stride, period and so on)
CPU board	COMMEL LP-170C (CPU: Intel Atom D525 1.8 GHz)
OS	Linux (Ubuntu12.04LTS)
Interface	Ethernet \times 1, USB \times 1, Speaker, DIP switch \times 4, Push switch \times 1
Servomotor	Futaba RS405CB \times 17
IMU	Analog Devices ADIS16375
USB camera	Shikino High-Tech KBCR-M05VU
Battery	3S (11.1 V, 5000 mAh)

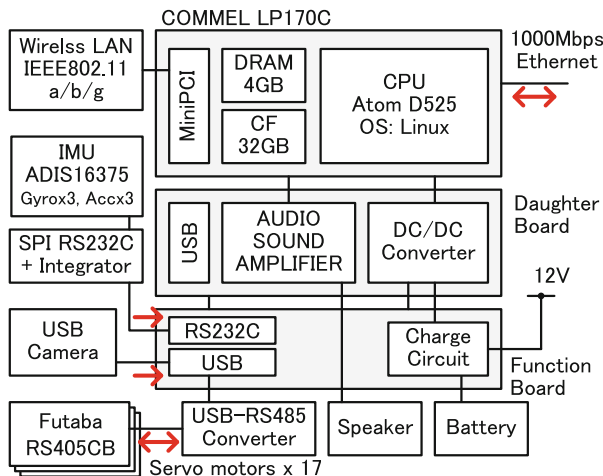


Fig. 3. Overview of the hardware system

3 Mobility

We achieved high-speed and stable mobility. The maximum speed of walk is approximately 0.4 m/s when carefully tuned. For playing soccer, it is also important to keep stable walking for long periods of time. Many soccer robots in the KidSize league tend to become unstable in the later part of games because of motor overheating. However, our robot can keep stable walk during full match. This is one of the key factors of our winning.

Gyro feedback is applied during walking to prevent falling down. Our robot does not usually fall down when it walks alone. However, in a soccer game, it often falls down when it is pushed by other robots. When our robot falls down, it detects the fall and stands up smoothly. The posture of the robot is estimated by fusing acceleration and angular velocity received from the IMU [15].

We designed a leg structure using parallel mechanism as shown in Fig. 4. This mechanism can keep pitch angles of the feet synchronized even if the motors are not synchronized completely. It contributes to the stable walking.

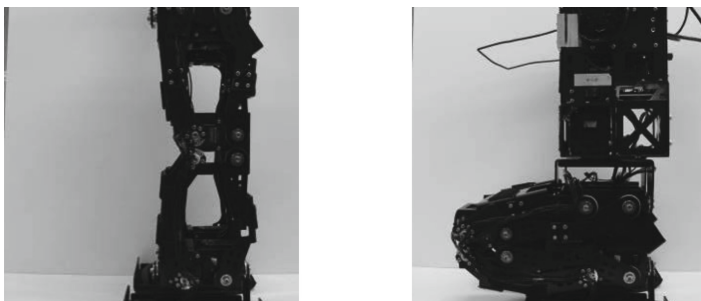


Fig. 4. Parallel mechanism of legs.

Choosing large torque servomotors is also important to prevent the overheating. We employ Futaba RS405CB with a maximum torque of 48 kg-cm. Active cooling fans are attached to motors for knee and ankle joints. Heat from the motors is suppressed by abovementioned factors, and therefore our robot can walk stably even after a long operation time.

4 Computer System

4.1 Hardware

The key advantages of the robot are a high computational capability and an ease of maintenance. A computer board we employ (LP-170C) has an Atom D525 CPU, which has substantially higher capability than DarwIn-OP's Atom Z530. We run Ubuntu Linux on the computer. All software modules we develop, including perception and control, are executed on it. Linux has many advantages in ease of installation and operation compared to other operating systems we have previously used (Windows and NetBSD). To improve the maintainability of electronic components, we designed a slot-in mechanism as shown in Fig. 5. The mechanism interfaces *function board* which supplies power and *daughter board* on which the main computer board is mounted. The pictures of the boards are shown in Fig. 6. By this mechanism we were able to eliminate a large number of cables compared with the previous model. The function board also has battery-charging functionality so that we can charge batteries without external chargers.

4.2 Development Environment

We have installed a development environment to each robot so that we can edit and compile source codes in the onboard computer. We directly operate the onboard computer by connecting a display and a USB keyboard (Fig. 7), or remotely operate it via VNC. The charging circuit significantly improved ease of development. While we are editing and compiling source codes, we plug the A/C adapter to the robot and charge the battery. Servomotors are automatically powered off when the A/C adapter is plugged. When we want to check software using the robot, we unplug the adapter then the power of the motors is automatically turned on and we only need to put the robot on the field. We achieved laptop-like usability.

5 Software System

5.1 Architecture

Figure 8 shows the architecture of the software system. All software modules including perception, planning, and control are executed on the main computer board. Two processes are executed on a single computer: one is for perception and planning and the other is for control. Images are captured by the USB camera, and processed on the

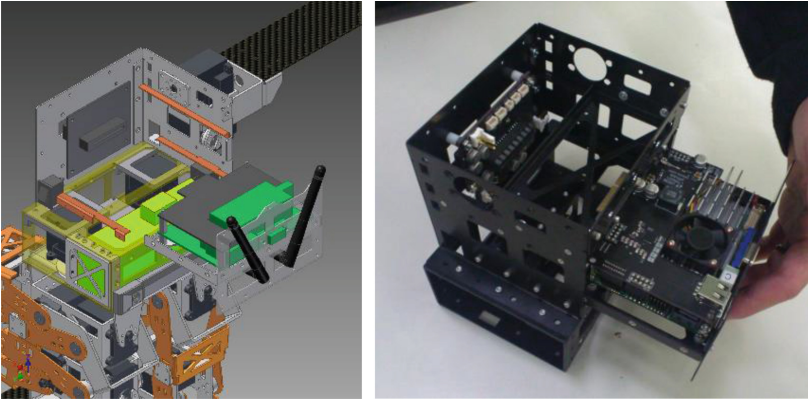
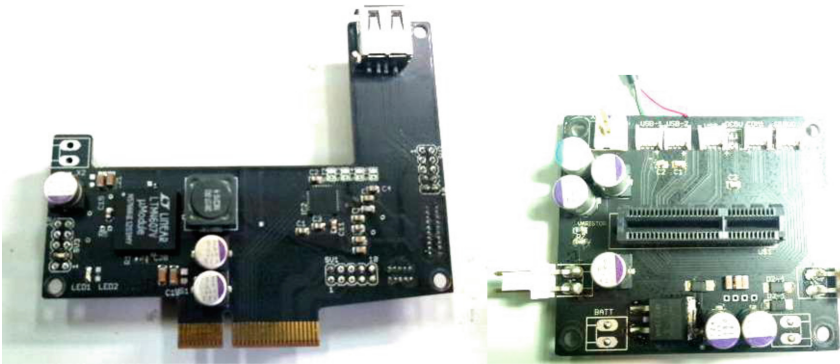


Fig. 5. Slot-in system



(a) Daughter board

(b) Function board

Fig. 6. Developed circuit boards

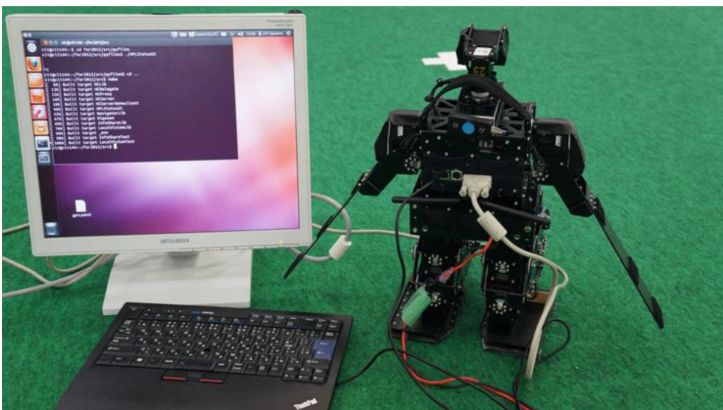


Fig. 7. Software development environment.

computer board to detect the position of the ball, other robots and landmarks. The robot continuously estimates self-position using the obtained information. Higher level of the robot behaviors such as following a ball are described in the soccer strategy programs. The soccer strategy programs are written in Python for ease of trial-and-error type of development while the rest of the software modules are written in C and C++.

The body control tasks are operated in a dedicated control process. It controls the posture of the robot according to commands sent from the main process. The status of the robot (e.g. posture) is periodically sent to the main process. The control process sends commands such as angles to servomotors. An IMU is used for gyro feedback and posture estimation.

We employ Internet Communication Engine (ICE) for communication among software components. ICE is a middleware for distributed computing including Remote Procedure Call (RPC). ICE is known to be computationally efficient compared to other middleware such as CORBA [16]. Our software modules running in different processes or computer hosts communicate with each other via ICE.

5.2 Image Processing

Objects such as a ball or robots are detected by color-based image processing. Object colors are detected using a pre-calibrated look-up table. A connected-component labeling algorithm then groups regions of the same color. Object positions are calculated from the object position in an image and the pose of the camera under the assumption that all objects are on the floor. The pose of the camera is calculated by inverse kinematics. The resolution of the images can be selected from 640×480 or 320×240 . Our algorithm runs at 20 fps with the onboard computer. The color look-up table must be calibrated before every trial. We developed a GUI (Fig. 9) to build the table efficiently. The operator registers colors in images by clicking the image on the GUI. The color detection result is also displayed and updated real-time; therefore the operator can interactively calibrate and verify the color table.

5.3 Localization

The robot position and orientation are estimated using a particle filter that fuses motion estimations and observations [17]. The hypotheses of robot position and orientation are represented by a set of particles $\{S^{(i)}\}_{i=1}^N$. Each time the robot moves, the new generation of particles are drawn according to the following proposal distribution:

$$S_t^{(i)} \sim P(x_t | S_{t-1}^{(i)}, \Delta x_t)$$

Here, Δx_t is the estimated robot motion from time $t = 1$ to t , which is calculated by the kinematics and integrating angular velocity from the IMU. The distribution is approximated by a normal distribution.

When landmarks such as white lines and goal posts are observed, the particles resampled according to the weight proportional to the observation likelihood

$$w^{(i)} \propto P\left(Z|s^{(i)}\right).$$

Errors of the landmark measurements are modeled by a normal distribution and correlation-based measurement model [18] is used for white line observations.

To deal with the localization ambiguity caused by the soccer field symmetry, the robot orientation is manually initialized when we put the robot on the field. The robot has an orientation reset button for this purpose and the robot handler pushes the button when the robot is manually repositioned during the game.

During soccer games, the localization system sometimes fails because of various reasons such as unexpected measurement errors and manual repositioning. To reduce the risk of own-goal, we integrated failure recovery algorithm proposed by Ueda et al. [19]. Our system keeps track of the measurement score and when the score falls below a certain threshold, the particles are diffused for re-localization.

5.4 Obstacle Avoidance

We use graph-based path planning for obstacle avoidance as illustrated in Fig. 10. Several control points are placed around the detected obstacles (i.e. other robots) and a complete graph is built by connecting all control points, start point, and destination point. Costs are set to all edges of the graph according to their length. Extra costs are added for edges that go across obstacles. Dijkstra algorithm is employed to find the path with minimal cost.

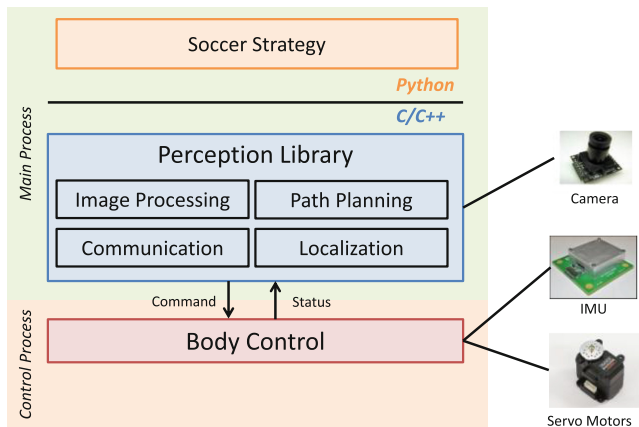


Fig. 8. Architecture of the software system.

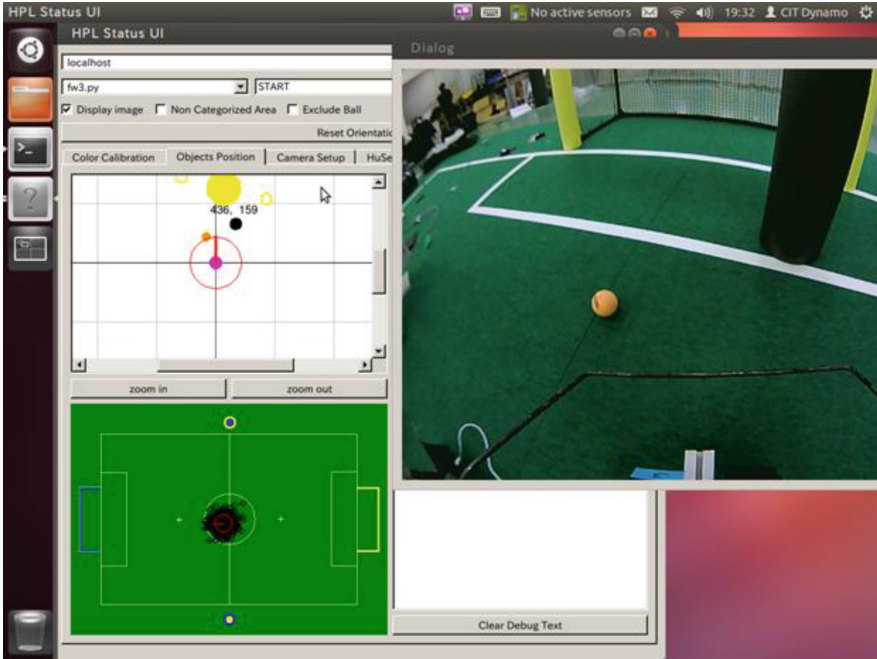


Fig. 9. Our graphical user interface. Positions of detected objects are visualized.

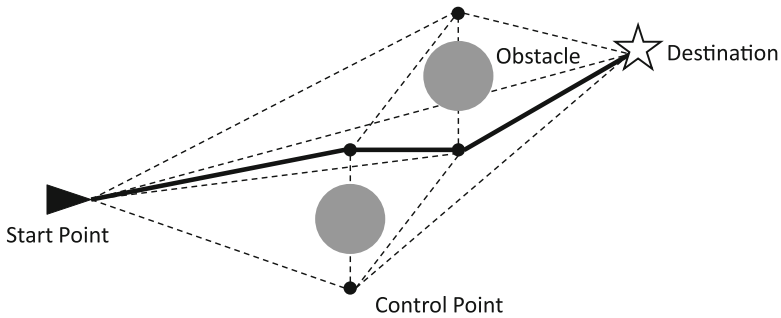


Fig. 10. Illustration of our path planning algorithm. Control points are placed around obstacles (small circles). Edges (dash lines) are generated to form a complete graph by connecting between start point, destination point and all control points. Dijkstra algorithm is employed to find the cost-minimizing path (bold line).

5.5 Tools for Software Development

We developed a user-friendly GUI tool for soccer strategy development environment (Fig. 11). Programmers can interactively check variety of items through this interface. The functionalities provided by the interface are as follows:

[User operation]

- (1) Send commands to the control process
- (2) Build a color look-up Table
- (3) Execute strategy programs by its name

[Status monitoring]

- (1) Image data (both raw and processed images)
- (2) Detected objects and their positions
- (3) Estimated robot position and particles
- (4) Debugging messages
- (5) Battery voltage and temperatures of servomotors

The GUI displays most of the significant status of the robot so that the programmer can check the algorithm and find problems easily.

We also developed a simulator based on V-REP. V-REP is an open source robot simulator made by COPPELIA ROBOTICS. Most of the behaviors of the robot are verified in the simulator. We check both the low-level walking control and high-level soccer strategy codes using the simulator and apply the verified codes to the real robot without any modification.

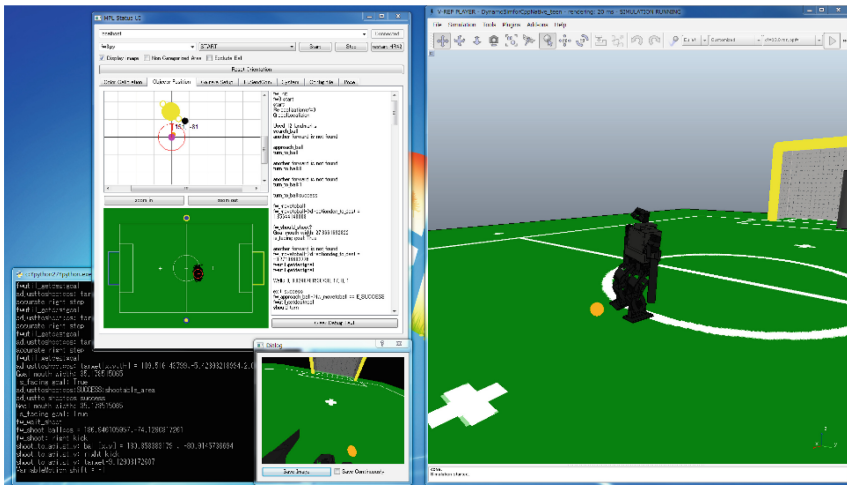


Fig. 11. Soccer strategy development environment and simulator software

6 Conclusion

In this paper we described our autonomous soccer humanoid system. Our system has high mobility, well-designed control system, position estimation by a monocular camera, user-friendly interface and simulator. High integrity of our humanoid system made it possible for us to win the Louis Vuitton Humanoid Cup at RoboCup 2014 Brazil.

References

1. Hayashibara, Y., Minakata, H., Sakamoto, H., Irie, K., Kaminaga, H., Fujita, S., Fukuta, M., Horiuchi, T.: Development of autonomous soccer robot system “CIT Brains and Hajime Robot” for RoboCup humanoid. In: Proceedings of JSME Robotics and Mechatronics Conference 2008, 2P1-J18 (2008)
2. Minakata, H., Hayashibara, Y., Ichizawa, K., Horiuchi, T., Fukuta, M., Fujita, S., Kaminaga, H., Irie, K., Sakamoto, H.: A method of single camera RoboCup humanoid robot localization using cooperation with walking control. In: The 10th International Workshop on Advanced Motion Control, TF-000787 (2008)
3. Hayashibara, Y., Minakata, H., Seike, Y., Ichizawa, K., Ogura, S., Irie, K., Sakamoto, H., Kaminaga, H., Kawakami, T., Fujita, S., Fukuta, M., Horiuchi, T.: Development of soccer robot system “CIT Brains”. In: Proceedings of JSME Robotics and Mechatronics Conference 2009, 1P1-C07 (2009)
4. Hayashibara, Y., Minakata, H., Sakamoto, H.: Autonomous soccer humanoid system “CIT Brains”. *J. Jpn. Soc. Des. Eng.* **44**(6), 334–340 (2009)
5. Hayashibara, Y., Minakata, H., Irie, K.: Development of humanoid “Dynamo” for RoboCup. In: Proceedings of the 28th Annual Conference of the Robotics Society of Japan, 1A3-5 (2010)
6. Hayashibara, Y., Minakata, H., Irie, K., Ichizawa, K., Tsuchihashi, I., Nohira, K., Sakamoto, H.: Development of a humanoid “Dynamo2011” for RoboCup. In: Proceedings of SICE System Integration Division Annual Conference 2011, 1C1-3 (2011)
7. Tsuchihashi, I., Minakata, H., Irie, K., Nozaki, K., Sakamoto, H., Hayashibara, Y.: Study on the influence of state transition of touch ground on a biped robot walking. In: Proceedings of JSME Robotics and Mechatronics Conference 2012, 2A2-Q06 (2012)
8. Tsuchihashi, I., Nozaki, K., Hayashibara, Y., Minakata, H., Irie, K., Sakamoto, H.: Study on the effect of joint backlash of a biped robot. In Proceedings of SICE System Integration Division Annual Conference 2012, 3N2-2 (2012)
9. Hayashibara, Y., Minakata, H., Irie, K., Sakamoto, H.: Development of a humanoid “Dynamo2012” for RoboCup. In: Proceedings of JSME Robotics and Mechatronics Conference 2013, 1P1-A12 (2013)
10. Nozaki, K., Tsuchihashi, I., Minakata, H., Irie, K., Sakamoto, H., Hayashibara, Y.: Study on the influence of state transition of touch ground on a biped robot walking (2nd report: consideration of the sole angle and the walking stability). In: Proceedings of JSME Robotics and Mechatronics Conference 2013, 1P1-A13 (2013)
11. Hayashibara, Y., Minakata, H.: The research development of autonomous soccer humanoids through the RoboCup. *J. Soc. Instrum. Control Eng.* **52**(6), 487–494 (2013)
12. Nozaki, K., Minakata, H., Irie, K., Sakamoto, H., Yoneda, K., Hayashibara, Y.: Study on the influence of state transition of touch ground on a biped robot walking (3rd report: relation between the roll angle of ankle joint and walking behavior). In: Proceedings of JSME Robotics and Mechatronics Conference 2014, 1P1-S03 (2014)
13. Fukuda, T., Minakata, H., Irie, K., Sakamoto, H., Hayashibara, Y.: Study on walking control of a humanoid robot considering motor temperatures (2nd report: relationship between walking patterns and motor temperatures). In: Proceedings of JSME Robotics and Mechatronics Conference 2014, 1P1-S01 (2014)
14. Sakamoto, H., Nakatsu, R.: Walking and running control of small size humanoid robot ‘HAJIME ROBOT’. In: The Thirteenth International Symposium on Artificial Life and Robotics 2008, GS6-1 (2008)

15. Madgwick, S.O.H., An efficient orientation filter for inertial and inertial/magnetic sensor arrays. Technical report, University of Bristol University, UK (2010)
16. Khan, S., Qureshi, K., Rashid, H.: Performance comparison of ICEHORB, CORBA and Dot NET remoting middleware technologies. *Int. J. Comput. Appl.* **3**(11), 15–18 (2010)
17. Dellaert, F., Fox, D., Burgard, W., Thrun, S.: Monte carlo localization for mobile robots. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1322–1328 (1999)
18. Thrun, S., Burgard, W., Fox, D.: *Probabilistic Robotics*. MIT Press, Cambridge (2005)
19. Ueda, R., Arai, T., Sakamoto, K., Kikuchi, K. Kamiya, S.: Expansion resetting for recovery from fatal error in monte carlo localization - comparison with sensor resetting methods, In: *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, pp. 2481–2486 (2004)