Proceedings of the 2004 IEEE
International Conference on Robotics & Automation
New Orleans, LA • April 2004

Integrated System Software for HRP2 Humanoid

Kei Okada Takashi Ogura Atushi Haneda Daisuke Kousaka Hiroyuki Nakai
Masayuki Inaba Hirochika Inoue
Graduate School of Information Science and Technology, University of Tokyo
#701, Engineering Building No. 8, 7-3-1, Hongo, Bunkyo-ku, Tokyo, Japan
k-okada@jsk.t.u-tokyo.ac.jp

Abstract— This paper describes design and development of system software for humanoid robots that researchers who specialized in not only biped walking but also various field are able to use humanoid robots as a research tool. For this purpose, the system for humanoid must integrate and organize each subsystems such as control, recognition, dialogue, planning and so on, and it must provide efficient full-body motion control method by specifying fewer degrees of freedoms than all joints. Our system design provides common interface among subsystems by implementing each function as a method call of tree-dimensional model of a robot for well integration, and it also provides motion planning technique based full-body posture sequence and walking pattern generation. Finally, we show integrated behavior experiments with vision, planning and motion control using developed system software for life-sized humanoid robot HRP2.

I. Introduction

Recently as the technology of developing humanoid hardware platforms have been progressed, the number of researchers who are interested in using robots as a research tool is increasing. However a special field of when is not only biped and humanoid robot but also vision, planning, dialogue, psychology and so on.

Therefore, system software for humanoid robots must integrate many subsystems to realize capability of not only executing pre-programmed motions, but also planning behaviors based on real world recognition and also interacting with people or environment. Moreover it also requires efficient full-body action control method that even non specialists are able to use a robot as a research tool.

For this reason, design of system that integrate and organize these subsystems well and efficient whole-body motion control capability is important to realize system software of humanoid robots.

Several research groups also have developed humanoid robots with control, vision and planning [1]–[4]. They argued that they developed integrated system, however, there are no discussion from a view of well-integration and well-organization, that is whether non specialists are able to use humanoid robot system software that covers many subsystems. We believe that integrated system software is not just a collection of subsystems but methodology of integration and organization of many subsystems that helps researches to develop high level behavior of a robot.

In this paper, we describe design of system software for humanoids that is; 1) Common interface among subsystems for well integration by implementing subsystems as a method call of a robot model. 2) Efficient full-body action control method by not specifying all joints of humanoid but only specifying a target location of a body or a target position of a link.

Fig. 1. Humanoid Robot HRP2

We also present developed system software that supports control, vision, auditory and planning for the HRP2, the first commercially available life-sized humanoid robots [5], [6]. The Department of Mechano-Informatics, Graduate School of Information Science and Technology, University of Tokyo introduced this humanoid robot as research and education tool in March 2003. In September 2003, six month later, our developed software enabled undergraduate students and master students to develop and research integrated behaviors along with the real humanoid robot as shown in Fig.1 . This experiment shows the usefulness of our developed system software.

The rest of this paper is organized as follows: Section 2 describes requirements and design principles of our integrated system software for humanoids, Section 3 describes the HRP2 humanoid robot, Section 4 shows our developed system software for HRP2, Section 5 presents some results using the HRP2 robot and Section 6 concludes with summary.

0-7803-8232-3/04/$17.00 ©2004 IEEE

II. Integrated System Software for Humanoids

This section describes design of system that integrate and organize subsystems well and efficient whole-body motion control capability is important to realize system software of humanoid robots. Followings are requirements.

Integrated and organized operation of subsystems

Integrated system software of humanoid robots should not be a collection of subsystems. Subsystems must be well integrated and organized. The system has to provide developers to feasible technique to connect subsystems each other.

Generate whole body behaviors easily

Programming of whole body behaviors is very complex because it must specify angles of all joints of humanoid at every time period. Therefore efficient motion control interface for generating whole body behavior is required.

Our design principles are as followings:

- Common interface among subsystems by implementing functions as a method call of three-dimensional model of a robot
  System architecture as shown in a left figure of Fig.2 , which functions of subsystems are called directory from programming environment, has no common interface of coordinates system or description of a body and environment between each subsystem. This makes developers to organize subsystems difficult, especially if software becomes large.
  Our approach is shown in a right figure of Fig.2 , which functions of subsystems are called through three-dimensional model of robots as a method call to realize common interface for subsystems and let developers to organize and integrate them efficiently.
- Efficient full body action controller by specifying a target body location or a target link position
  In order to control a robot, commands to all joints at each time sequence are necessary, however, in the case of humanoids with have many joints, an efficient action control method by specifying fewer degrees of freedom than all joints makes researchers to develop behaviors efficiently. Our integrated system software provides motion planning based action control method, which generate full body motions from given a target body location or a target link position.
- Interactive software development using an interpreter language
  In order to modifying programs while a robot is moving, computer languages that needs compile requires robots to stop and start again whenever developers modify programs. We adopt interpreter language as programming development environment.

III. HRP2 Humanoid Robot

The HRP2 humanoid robot is the final version of humanoid robotics platform of HRP(Humanoid Robotics Project) [5], [6], which is sponsored by MITI/NEDO and runs from 1998FY to 2002FY. HRP2 is 154[cm] height, 58[kg] weight, 30 DOF includes 2DOF for waist, contains computers and butteries in its body.

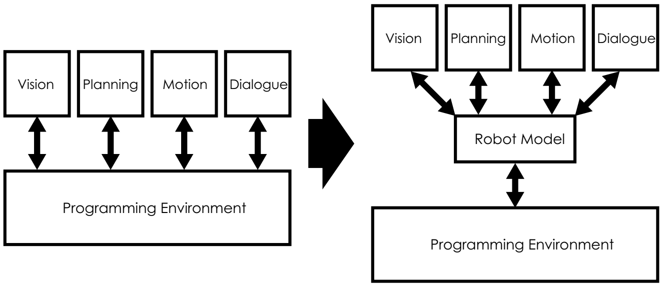

Fig. 2. Common interface among subsystems by implementing functions as a method of three-dimensional model of a robot

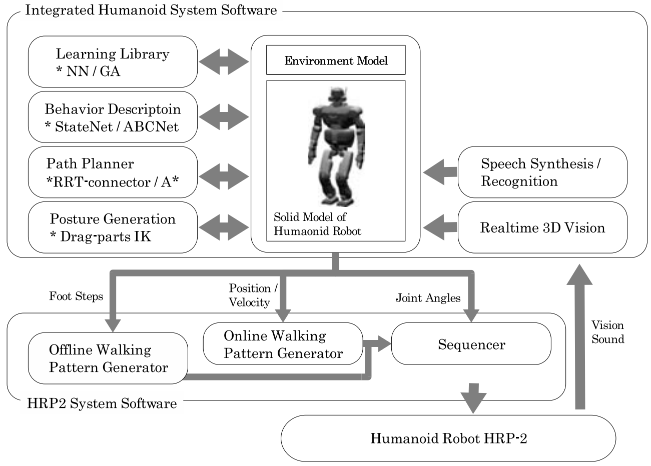

Fig. 3. Integrated system software for humanoid robot HRP2

A. HRP2 control system

HRP2 control system [7] includes walking and controlling facilities, is written on a OpenHRP [8] architecture.

In the HRP2 control system, walking and controlling functions are implemented as plugins. Each plugin runs at every control loop(5[msec]) sequentially and output to a robot through the adpter.

B. Motion control in HRP2 control system

From a motion control view, functions in the HRP2 control system are classified into a whole body posture sequencer level and a walking pattern generation level. In the former level, users are able to command a target posture of a robot body, In the latter level, users are able to command a target position, a target velocity, or a footstep locations for walking control. Therefore, integrated system software for HRP2 requires to output whether sequencer level or walking pattern level described above.

IV. Integrated System Software for HRP2 Humanoid

This section describes the integrated system software for the humanoid robot HRP2(Fig.3), which consists of vision, auditory, planning, and walking control software.

We employ EusLisp [9], whose facilities includes interactive programming, multithread, three-dimensional solid modeling, foreign function interface, as a programming environment for the system software.

A. Three dimensional model of robots and environment

We have build three-dimensional model of robots and environment by using solid modeling facility of EusLisp. HRP2 robot is implemented as a sub-class of robot class. For instance, modification of a posture of a robot is realized by sending a message through a method call to the instance of HRP2 robot.

Three-dimensional geometric, kinematical and dynamics model of HRP2 robot is created by parsing VRML data which is provided with HRP2 control system. We also utilize simplified version of the model by calculating bounding box of each link.

B. Loading libraries using foreign function interface

In order to utilize foreign libraries such as visual processing libraries which computation speed is required or commercial auditory library in EusLisp environment, we employ foreign function interface facility for loading them.

We have developed C++ based robot vision library for in-house purpose. From EusLisp environment, interfaces for setting parameters, processing, getting result are implemented. ViaVoice and SpeechSDK are also loaded to provide auditory functions.

C. EusLisp and HRP2 control system via CORBA

EusLisp and HRP2 control system are connected through CORBA architecture [10]. CORBA is vender, OS and language independent distributed object architecture that to work together over network.

In the HRP2 control system, posture controlling and walking functions are provided by plugins and controlled by a plugin manager. Plugins and a plugin manager are implemented as CORBA objects. Thus, these functions are able to called from other software. We have implemented following facilities.

- A plugin manager has a stdin/out interpreter for accepting commands. We added CORBA interface for sending message from EusLisp to command interpreter of a plugin manager.
- We added a plugin for providing sensory data of a robot to EusLisp. This plugin sends joint angle, force sensor, gyro sensor, and accelerometer data via CORBA interface.

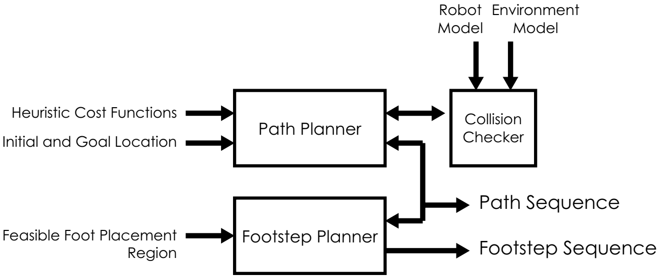

Fig. 4. An overview of path planning system

V. Action Control of Humanoids using Motion Planning

In order to control a robot, it is necessary to command angles of all joints at each time sequence. In the case of humanoid robots with many joints, a method to generate full body motion trajectory of a robot by specifying fewer degree of freedoms is required.

We implimented an efficient action control method that generates motion of a robot by specifying a target body location and a target position of an arbitrary link(s) by using a path planning technique and a full body planning technique.

A. Motion planning for navigation

Research on motion planning has large numbers of literatures, especially for wheeled robots. On the other hand, A few for legged robots including humanoids, these are categorized as path planning [11], [12] or footstep planning [13], [14]. In contract to these systems, our system provides both techniques so that developers are able to choose appropriate algorithm.

Fig.4 shows an architecture of our path planning system, first, it generate path from initial location to goal location, then generate footstep sequence along with the path.

1) Path planning: Our path-planning algorithm accepts initial and goal location and heuristic const functions as an input, and generates a collision-free path trajectory as a output. In this planner, the body of the robot is modeled as a bounding box or a cylinder. We adopt the A* search algorithm for planning.

2) Footstep planning: An algorithm to plan footstep sequences from a given path is as follows [15]:

(1) For the given path of the robot.
(2) Calculate the virtual swing leg path along with the given path.
(3) Calculate the point where the virtual swing leg path and a boundary of the feasible foot placement regions are intersect. Set this point as the foot placement point of the swing leg.
(4) Switch the swing leg and the support leg, iterate (2) (3) until the robot reaches to the goal location.

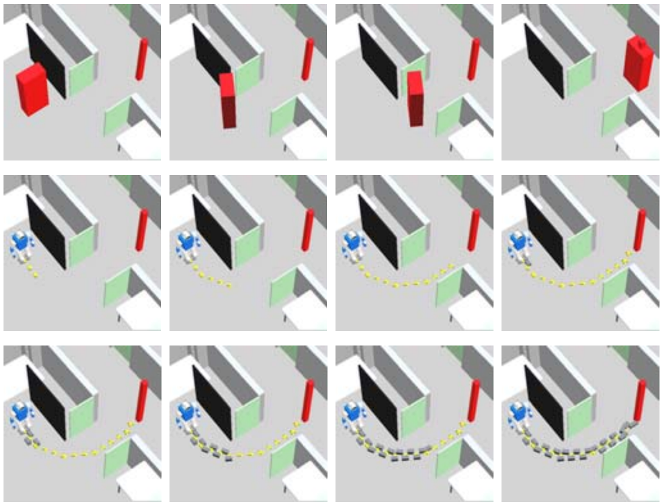

Fig. 5. Experimental results of path planning: Top) Path planning using bounding box of a robot, Middle) the results of path planning, Bottom) footstep planning along with the path.

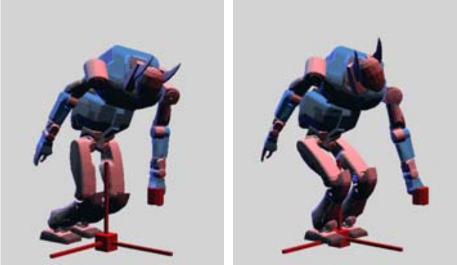

Fig. 6. Statically-stable full body posture planning

Fig.5 shows results of planned path from a given starting position and goal position. Our system is able to generate both a trajectory and foot prints as a results.

B. Motion planning for full body posture sequence

Full body planning algorithm accepts a target position of arbitrary link(s) and pinned link(s) such as legs, and generates collision free full body posture sequence motion. Therefore, this algorithm is used for generating a reaching posture from a given target position. In contract to previous full body reaching researches on humanoids, which only considers an upper body or a heuristic approach [16], [17], our approach is so general that is able to apply to many tasks.

In order to realize full body planning, we adopt inverse kinematics algorithm with various types of constraints. [18]. This algorithm has capable of calculating full body posture by the given target position of the arm while both legs are pinned to the ground.

We extend this method to statically-stable full body posture generation method [19], and collision-free motion generation method [20].

1) Full body posture planning under constraints: The inverse kinematics of full body motion under two priority constraints algorithm [18] finds the solution that satisfies the lower priority constraint by looking for it in the null-space of the higher priority constraint. If there is a conflict among the constraints, a least-square method is utilized for finding the approximation.

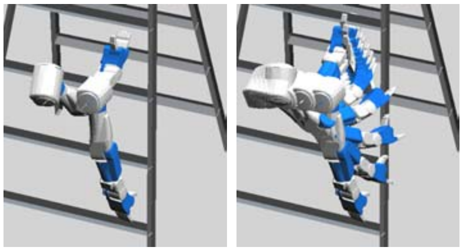

Fig. 7. Collision-free full body motion planning

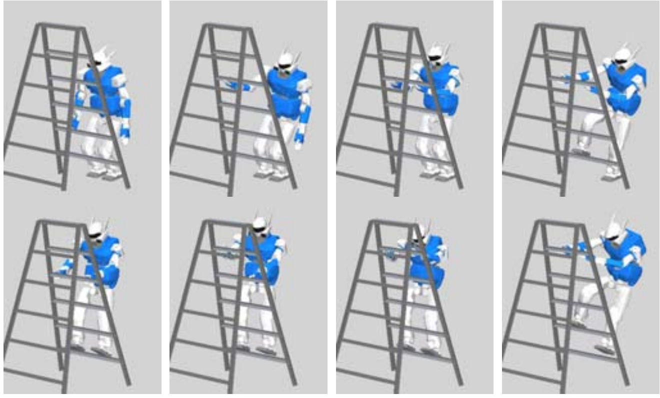

Fig. 8. Climbing ladder experiment using collision-free full body motion planning

2) Statically-stable full body posture planning: In order to apply the results of full body posture planning to real robots, stability of a robot is important. We extend the full body posture planning method to satisfy static stability by updating a constraint in order to keep position of a center of gravity within the foot sole polygon.

Fig.6 shows the experimental results of the proposed algorithm. The task of this robot is to reach the left hand to the target. The left figure is generated posture without stability compensation and the right is the result of the statically stabile planning.

3) Collision free full body motion planning: In this algorithm, the robot is divided into movable limbs, fixed limbs, free limbs. First, it determines target posture using full body posture planning, then generate a motion sequence of movable limbs by using motion planning technique. Motion planning for a robot body provides high dimensional search space, therefore we adopt RRT connect planner [21], then generate motions of fixed limbs.

Fig.7 shows the results of collision-free full body motion planner. Fig.8 shows the result of climbing ladder behavior using this algorithm.

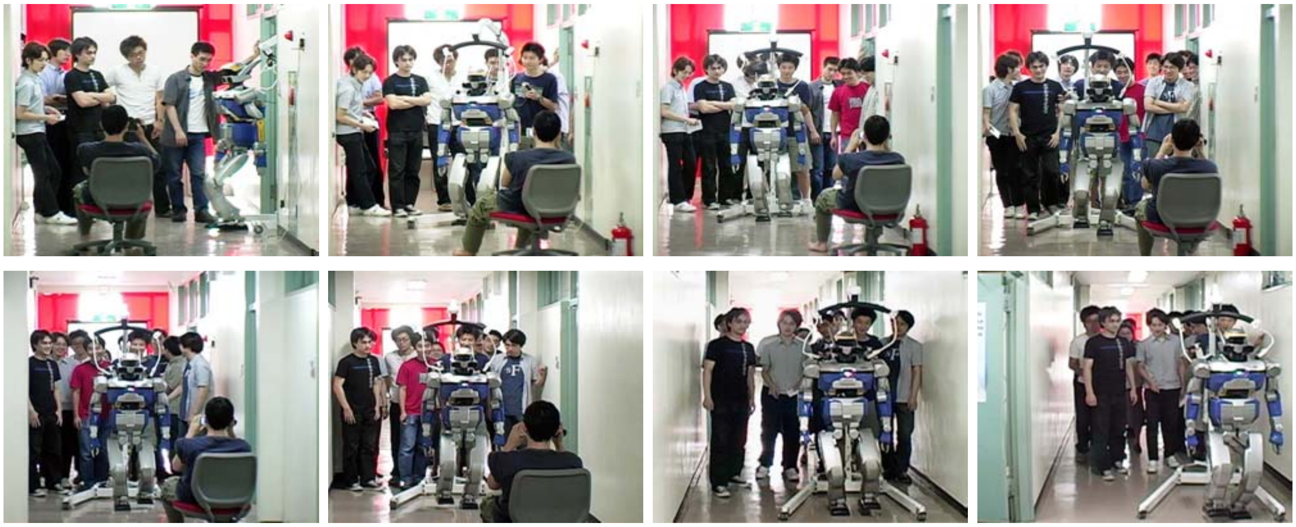

Fig. 10. Navigation experiment based on path planning using environmental model

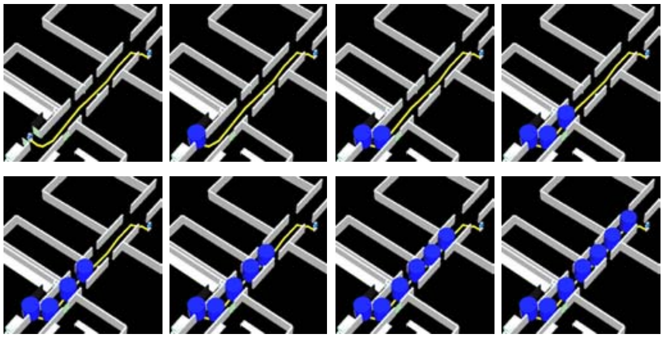

Fig. 9. Path planning using environmental model

VI. Integrated Behavior Experiments using HRP2

A. Navigation experiment based on path planning using environmental model

We build an environmental model of our laboratory includes seven rooms and a corridor. In this model, we set a initial location in one room and a goal location in other room that is 16[m] away. Fig.9 shows the path planning results using this environment. Fig.10 shows results using the real humanoid robot HRP2.

B. Whole body reaching experiment using statically stable full body motion planning

For the whole body reaching experiment, we build four postures which is initial, look down, reaching, final posture as shown in Fig.11 . While look down posture, the position of the target is calculated using a stereo vision processing, then generate the reaching posture using the full body motion planning technique by setting pins at the both legs and dragging the left hand to the target position. Fig.12 shows input images and results of correlation detection and depth map generation

Fig.13 shows simulated behavior using OpenHRP and Fig.14 shows an experiment using the real humanoid robot.

VII. Conclusion

The contributes of this paper are summarized as follows:

1) We presented the design of integrated system software for humanoid robots. In contract to the other integrated system that is usually a collection of subsystems. Functions of subsystems are implemented as method calls of a robot model. This feature provides common interface between subsystems so that developers can integrate and organize subsystems efficiently.
2) We propose efficient action control method for humanoid robots as the key feature of integrated humanoid robots, that generates motions of whole body of a robot by specifying a target body location or a target position of arbitrary links. This makes researcher to develop vision, auditory or planning based high-level behaviors easily.
3) We presented an algorithm for computing statically stable collision free trajectories given a target position of arbitrary link by combination of inverse kinematics method and RRT connect path planner.
4) We demonstrated examples of experimental research using life-sized humanoid robot HRP2. One is a navigation behavior based on a path planning using the environmental model of our laboratory. Another is a whole body reaching behavior using whole body motion planning. These experiments carried out by undergraduate and master students. It shows the usefulness and the efficiently of our developed system.

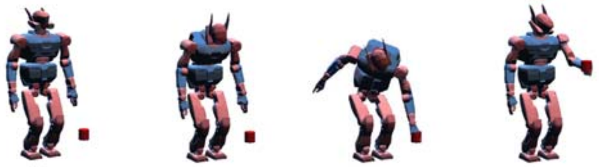

Fig. 11. Whole body reaching posture sequence

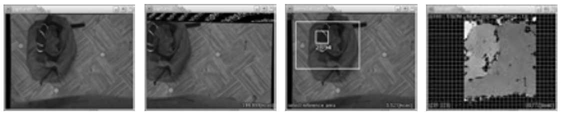

Fig. 12. Calculate the position of the target using visual processing

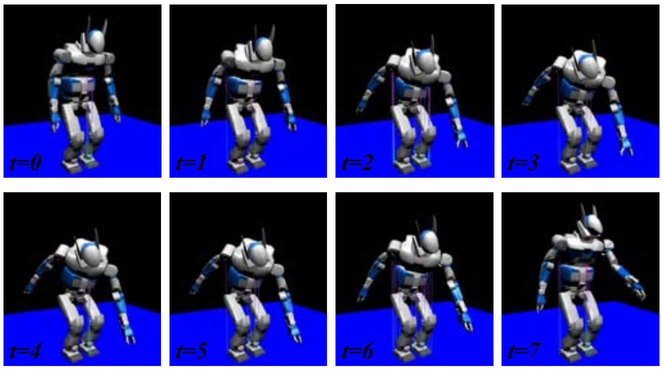

Fig. 13. Whole body reaching experiment using the simulator

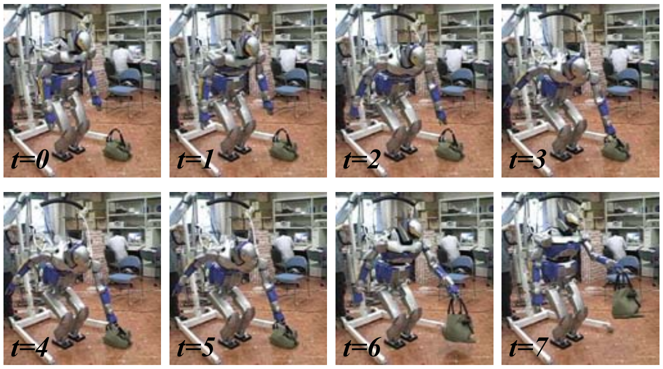

Fig. 14. Whole body reaching experiment using the HRP2 robot

References

[1] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura. The intelligent ASIMO: System overview and integration. In Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'02), pages 2478–2483, 2002.

[2] K. Sabe, T. Ohashi, K. Kawamoto, J.S. Gutmann, M. Fukuchi, and T. Yoshigahara. Obstacle Avoidance and Path Planning for Humanoid Robot using Stereo Vision. In National Symposium on Sensing via Image Information (SSII'02), 2002. (In Japanese).

[3] J.F. Seara and K.H. Strobl and G. Schmidt. Path-Dependent Gaze Control for Obstacle Avoidance in Vision Guided Humanoid Walking. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 887–892, 2003.

[4] S. Kagami, K. Nishiwaki, J.J. Kuffner, Y. Kuniyoshi, M. Inaba, and H. Inoue. Online 3d vision, motion planning and bipedal locomotion control coupling system of humanoid robot : H7. In Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'02), pages 2557–2562, 2002.

[5] H. Inoue, S. Tachi, K. Tanie, K. Yokoi, S. Hirai, H. Hirukawa, K. Hirai, S. Nakayama, K. Sawada, T. Nishiyama, O. Miki, T. Itoko, H. Inaba, and M. Sudo. HRP: Humaonid Robotics Project of MITI. In Proceedings of the First IEEE-RAS International Conference on Humanoid Robots (Humanoids 2000), 2000.

[6] H. Hirukawa, F. Kanehiro, K. Kaneko, S. Kajita, K. Fujiwara, Y. Kawai, F. Tomita, S. Hirai, K. Tanie, T. Isozumi, K. Akechi, T. Kawasaki, S. Ota, K. Yokoyama, H. Honda, Y. Fukase, J. Maeda, Y. Nakamura, S. Tachi, and H. Inoue. Humanoid Robotics Platforms developed in HRP. In Proceedings of the 2003 IEEE-RAS International Conference on Humanoid Robots (Humanoids 2003), 2003.

[7] General Robotix Inc. http://www.generalrobotix.com.

[8] F. Kanehiro, K. Fujiwara, S. Kajita, K. Yokoi, K. Kaneko, and H. Hirukawa. Open Architecture Humanoid Robot Platform. In Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA '02), pages 24–30, 2002.

[9] T. Matsui. Multithread object-oriented language euslisp for parallel and asynchronous programming in robotics. In Workshop on Concurrent Object-based Systems, IEEE 6th Symposium on Parallel and Distributed Processing, 1994.

[10] Object Management Groups. http://www.omg.org.

[11] K. Okada, M. Inaba, and H. Inoue. Integration of real-time binocular stereo vision and whole body information for dynamic walking navigation of humanoid robot. In Proceedings of International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI'03), pages 131–136, 2003.

[12] K. Sabe, T. Ohashi, K. Kawamoto, J.S. Gutmann, M. Fukuchi, and T. Yoshigahara. Obstacle avoidance and path planning for humanoid robot using stereo vision. In Proceedings of National Symposium on Sensing via Image Information (SSII'02), 2002 (In Japanese).

[13] J.J. Kuffner, K. Nishiwaki, S. Kagami, M. Inaba, and H. Inoue. Footstep planning among obstacles for biped robots. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'01), pages 500–505, 2001.

[14] O. Lorch, A. Albert, J. Denk, M. Gerecke, R. Cupec, J. F. Seara, W. Gerth, and G. Schmidt. Experiments in vision-guided biped walking. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'02), pages 2484–2490, 2002.

[15] T. Ogura, K. Okada, M. Inaba, and H. Inoue. The United Behavior Planning System for Integrated System Software on HRP-2. In Proceedings of the 21th Annual Conference of Robotics Society of Japan, page 2A13, 2003 (in Japanese).

[16] K. Inoue, Y. Nishihama, T. Arai, and Y. Mae. Mobile manipulation of humaonid robots – body and leg control for dual arm manipulation –. In Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA '02), pages 2259–2264, 2002.

[17] M.Inaba, T.Igarashi, S.Kagami, and H.Inoue. A 35 dof humanoid that can coordinate arms and legs in standing up, reaching and grasping an object. In In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 29–36, 1996.

[18] K. Yamane and Y. Nakamura. Synergetic CG Choreography through Constraining and Deconstraining at Will. In Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA '02), pages 855–862, 2002.

[19] K. Okada, T. Ogura, M. Inaba, and H. Inoue. Vision based Whole-body Motion using Integrated System Software on HRP-2. In Proceedings of the 21th Annual Conference of Robotics Society of Japan, page 2A22, 2003 (in Japanese).

[20] H. Nakai, M. Inaba, and H. Inoue. Whole Body Motion Generator of Robots using RRT-Connect Planner. In Proceedings of 2003 Annual Symposium of Robotics-Mechatronics (ROBOMEC '03), pages 2P2–1F–C1, 2003 (in Japanese).

[21] J.J. Kuffner and S.M. LaValle:. Rrt-connect: An effcient approach to single-query path planning. In Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA '02), pages 995–1001, 2002.