CIT Brains (Adult Size League)

Hideaki Minakata[1], Yasuo Hayashibara[1], Kiyoshi Irie[1],
George Tsukioka[1], Joshua Supratman[1], Moeno Masuda[1], Yasufumi Suzuki[1],
Daiki Maekawa[1], Kazuyoshi Makisumi[1], Masayki Ando[1], Ryu Yamamoto[1],
Shouta Hirama[1], Yukari Suzuki[1], Chisato Kasebayashi[1], Akira Tanabe[1],
Youta Seki[1], Yuya Hirata[1], Yuuki Kanno[1], Tomotaka Suzuki[1], Ryoma Aoki[1],
Akihisa Iwasaki[1], Fumihiro Ueki[1], Ryohei Goto[1],Satoshi Shimada[1],
Rena Shimoda[1], Takaharu Nakajima[1], Riki Hayashi[1], Yoshiaki Yuki[1],
Shigechika Miki[2], Yoshitaka Nishizaki[3], Kenji Kanemasu[4],
Hajime Sakamoto[5]

[1]Chiba Institute of Technology, 2-17-1 Tsudanuma, Narashino, Chiba, JAPAN
hideaki.minakata@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

Abstract. In this paper, we describe our system for the RoboCup soccer adult size humanoid league. The system we developed has high mobility, well-designed control system, position estimation by one camera and user-friendly interface. The robot can walk speedy and robustly. It also has a feedback system with gyro sensor to prevent falls. The robot has two control boards. One is for body control and another is for image recognition, behavior determination and so on. The latter CPU board is a standard PICO-ITX embedded personal computer. The robot detects the positions of landmarks by image processing. From the positions, the robot can also estimate own position by using a particle filter.

Keywords: high mobility, user-friendly interface, education tool

1 Introduction

In this paper, we describe on our system for the RoboCup soccer adult size humanoid league. Our robot is well designed and controlled robustly. Our team members are specialists from several technological areas. We integrate our technologies for developing an intelligent humanoid robot. Hajime Research Institute developed the mechanism and control system of the robot. Chiba Institute of Technology developed computer system and overall intelligence such as image recognition and soccer algorithm. Remarkable topic is that the most of members are undergraduate students. Through this development, the professors try to make an educational and research platform robot system of intelligent humanoid. The undergraduate student programs almost all of behavior decision algorism heuristically.

We developed two size robots for RoboCup. One is for kid size league and another one is for teen size league. The kid size robot “accelite” received 1st prize in RoboCup 2014 and 2015. The teen size robot “xega” received 2nd prize in RoboCup 2009, 2010, 2012, 2013, and also received 1st prize for technical challenge in 2010. Our team is planning to make brand new robots for Robocup2016. We are also planning to attend adult size league. Therefore we modified the teen size robot to the adult size this year to gather the data of human-size robot. The new leg and neck parts were made for this size-up.

Mainly, Hajime develops the robots with the same control board. Therefore, we can control different size robots with same command system. It enables to decrease the cost to develop the system. We can apply almost same program to them. Furthermore, the robots can perform with high mobility and stability. The normal speed is approximately 0.4m/s. It can also walk to any direction and angle smoothly. For stable walking, it has gyro sensor. The robot has two CPUs. One is used for body control, and another is used for image recognition, behavior determination and so on.

2 Overview of the System

The photograph of our robot is shown in the Fig.1. The specification of the robot is indicated in the Table 1. The overview of the control system is shown in Fig. 2. Our robot system consists of a camera, computers, sensors, servomotors, batteries and some user interfaces such as switch and LED. The camera sends image signal to the main CPU board. The signal is captured and stored in frame buffer memory. The CPU processes the image data to detect positions of ball, robots and landmarks. From the landmarks’ positions, the robot estimates own position by using a particle filter. From these data, the robot selects a next behavior. The behaviors that we can choose are not only just simply moving, but also complex behaviors such as following the ball. The action command is sent to sub CPU via RS232C network. The CPU decodes and executes the command. It sometimes returns the status data to the main CPU. If the command is a kind of moving the body or checking a status, the sub CPU sends a command to servomotor via RS485 network. Each servomotor has own CPU to control motor and receive/send commands. Because all servomotors are daisy-chained, the command is sent to all motor. The command includes ID number, so the servomotor can identify the command to which is sent. The servomotor decodes and executes the command. The displacement angle is controlled in local motor unit. The sub CPU should not send commands at short intervals. In total, this system is constructed as a well-designed hierarchic system. Therefore, we can modify the system easily.

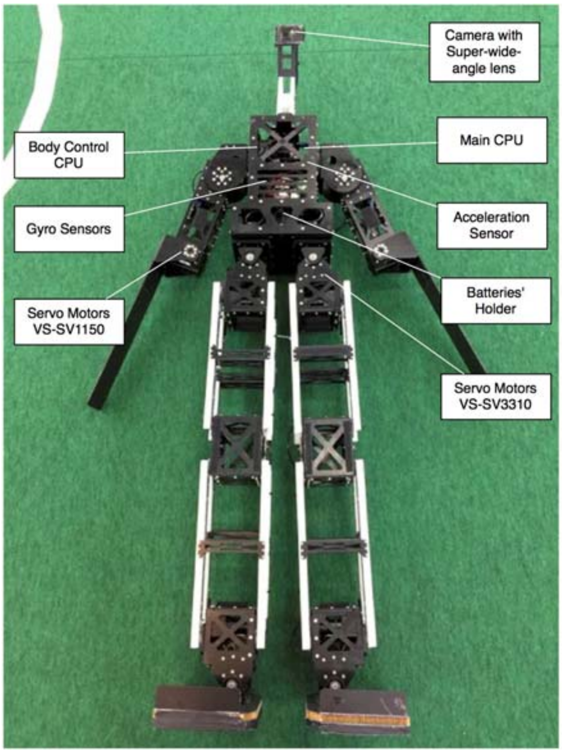

Fig.1. Structure of the Robot

Table 1. Specification of the Robot

Weight	13.0 kg (Including Batteries)
Height	1400 mm
Velocity (Forward)	0.4 m/s (maximum)
Walking Directions	All direction and rotation (Select the angle, stride, period and so on)
CPU Board	Main: COMMEL LP-170 (Intel Atom D525 1.8GHz) Sub: Hajime Robot HC5 (Renesas SH-2A/7211)
OS	Windows 7
Interface	Ether(100Base-TX) x 1, USB x 1(USB-wireless LAN), CF x 1, RS232C x 2, Sound In/Out, Digital I/O, etc
Servo Motor	Vstone VS-SV3300 x 2, VS-SV3310 x 4, Hajime Servo x10, ROBOTIS RX-28 x1
Battery	Li-Po 11.1V x 1

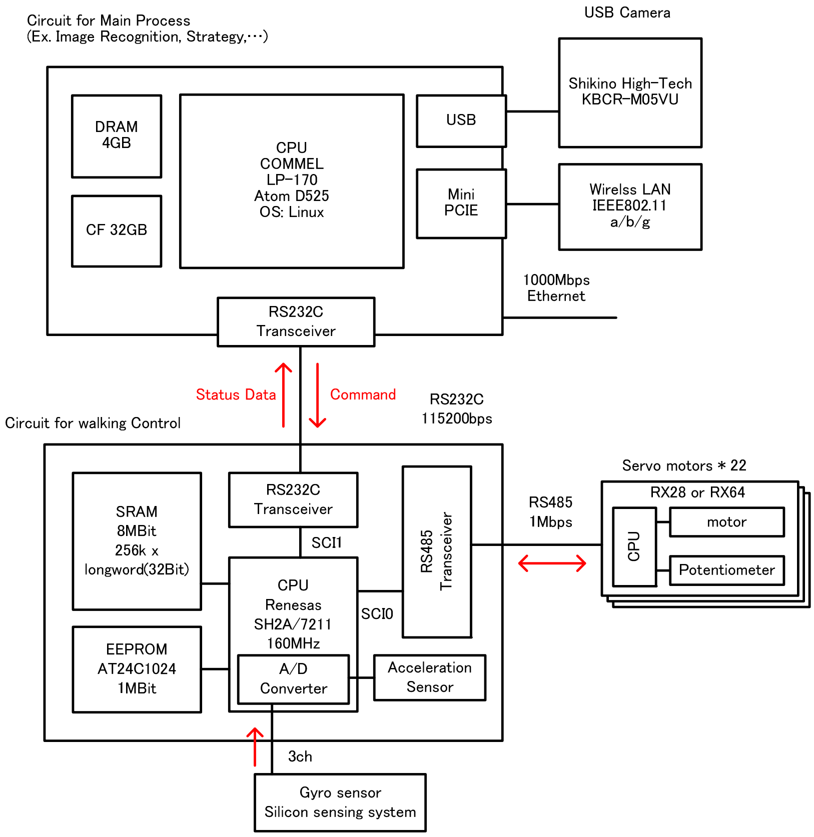

Fig.2. Overview of the Control System

3 Mobility

We apply parallel links to the legs. Through the mechanism, the stability of walk becomes much better thus, the number of servomotors also decreased compared to the previous robot. At the neck and shoulder, energy-absorbing mechanism is applied. Even if the robot falls, the camera and body could not break.

4 Computer System

One of significant feature of the robot is the high computational capability. The robot is capable of processing VGA (640x480) images 20 frames per a second. The CPU is Atom D525 and the operating system is Microsoft Windows 7. It processes the image data, estimates the positions and determines the behavior. After these processes, it sends a command to sub CPU board for controlling the robot. Moreover, it was possible to develop easily by adopting Linux that was accustomed to the operation and installing the development setting.

5 Image Processing and Position Estimation

As mentioned above, the computer processes the image data of 20 frames per a second. The resolution of the camera can be selected from 640x480 and 320x240. By simple image processing, it can detect the region of the same color. According to those data, it calculates the positions of ball, robots and landmarks. The position and direction of camera is calculated by inverse kinematics. The result is send and displayed on a PC. An example of the calculation is shown in Fig.3. Before this image processing, we should input the threshold of the color. We made an interface to input the value smoothly. The operator can change the value on GUI interface and check the effectiveness of the values immediately.

By measuring the positions of landmark, the position of the robot is estimated. We apply a particle filter to estimate it. It is shown in Fig. 4. If the robot detects the landmarks, the particles gather and bundle to collect position like the figure. The accuracy of the estimated position is not enough the goalkeeper to move home position. Thus we are now trying to detect the white line to reduce the position error.

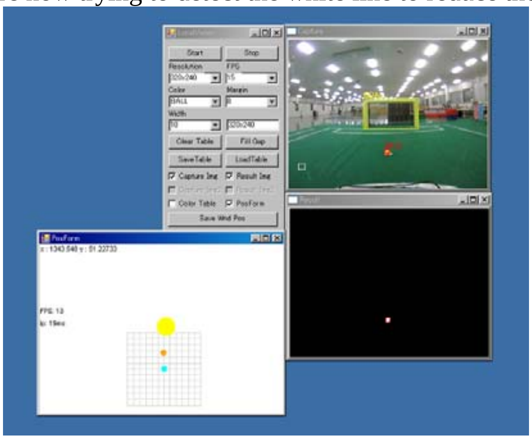

Fig.3. Graphical User Interface

Fig.4. Estimating Process Using Particle Filter

6 Ball Approach

Ball approach has been improvised to imitate closely to a humanistic movement. Our past method of ball approach is for the robot to approach the ball in the quickest way possible then to reposition itself toward the goal as shown in figure 5. We instead improvised our robot to reposition itself toward the goal while approaching the ball in the quickest way possible as shown in figure 6.

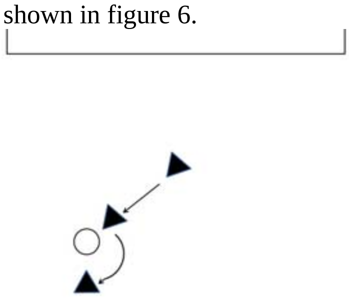

Fig.5. Original Ball Approach

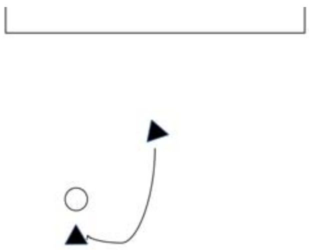

Fig.6. Improvised Ball Approach

7 Soccer Strategy

We are currently developing a new type of AI strategy using Hierarch Task Network (HTN) planner. In the past our AI strategy was structured using finite state machine (FSM). The FSM however, have several limitations such as its inability to express states as a plan. In contrast, the HTN planner is a planning architecture that expresses problems as a higher level task. Tasks are decomposed through a planning process which ultimately leaves with a series of atomic task. In this way, the robot's action can be planned ahead with the given situation. Therefore to overcome the limitation of FSM we structured our AI strategy using HTN planner. Figure 7 compares the two AI architectures.

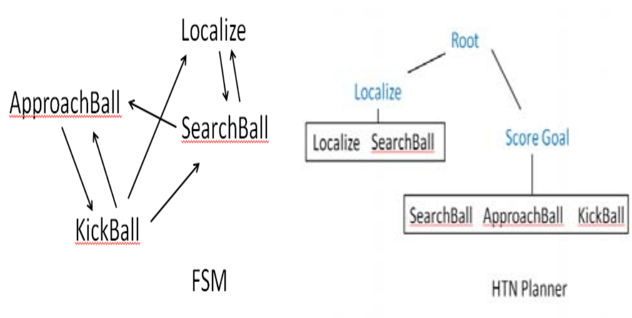

Fig.7. Comparison Between AI Architecture

8 Strategy Development Environment

We develop a user-friendly interface for strategy development environment. The programmer can check many things in this interface. The interface is provided as following.

[output]

1) simple command to sub CPU (the command can also generate through joystick, mouse and keyboard)
2) threshold of color (its effectiveness can be check immediately)
3) strategy name like forward and keeper (it select the program in robot)
4) fight side and our color

[input]

1) image data (It is possible to display the result of image processing)
2) detect and estimate positions (It is indicated graphically and saved in storage.)
3) command to sub CPU (We can check the algorithm)
4) message (If the programmer want to know the robot status, they can insert the message in the program. It is also saved in storage)
5) color values (We use the YUV color value.)

These are examples of input/output data. More data is interacted on this interface. Using this interface, the programmer can check the algorism easily. He/She can refer almost all data, so he/she can find the problem smoothly.

9 Conclusion

In this paper we described on our system. Our system has high mobility, well-designed control system, position estimation by one camera and user-friendly interface.