ZJUDancer 2009 Team Description

Tang Qing Tao Yikun Xiang Yuanfei
Xiong Rong
State Key Laboratory of Industrial Control Technology,
Zhejiang University, Hangzhou, China.
Email: tangq@iipc.zju.edu.cn

Abstract

This document describes the RoboCup Humanoid League team ZJUDancer from Zhejiang University, China, as required by the qualification procedure for the competition to be held in Graz, Austria, in July 2009. Full details of our robot including mechanical design, electrical design, sensor equipped and software design are described. With this robot, we hope we could get a much better result in 2009.

1 Introduction

The robots developed by ZJUDancer for RoboCup 2009 are fully autonomous humanoid robots which play different parts as a team in the football game. ZJUDancer is established in July, 2006 to develop humanoid robot in Zhejiang University. Every year, lots of students from varieties of colleges gathered together to join our team. We developed 3 generations of humanoid robot with which we won the champion of RoboCup 2007 China Open. Now, we look forward to the RoboCup 2009 Graz. Fig. 1 shows our robot in the

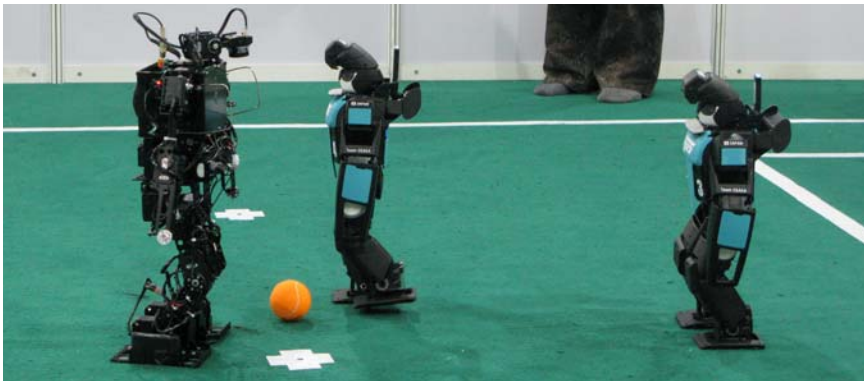

Figure 1: RoboCup 2008 Suzhou, VS TeamOSaka

Table 1: General Specifications of the robot

Team Name	ZJUDancer
Robot Name	Wukong, Bajie, Shaseng
Number of Freedom	20
Height	55cm
Width	35cm
Weight	3kg

competition against TeamOsaka in RoboCup 2008. The rest of the paper describe the details in mechanical design, electrical design, sensor equipped and software design of our robot.

1.1 General specification of the robot

Table. 1 shows the general specifications of our robots. Three players from ZJUDancer named by Wukong, Bajie and Shaseng are fully autonomous humanoid soccer robots. Each robots are fixed to the size and weights limitations of the competition and connected by wireless networks. Referee's directions could be sent to the robot through the network. More details could be introduced in the following sections.

2 Mechanical Specifications

The robot from ZJUDancer has 2 legs, 2 arms, 1 trunk and 1 head. The actuator we choose is Dynamixel RX-28. The reason that we choose this motor is that the RX-28 offers a high torque with less weight. Each robot is driven by 20 servo motors: 6 per leg, 3 in each arm and 2 in the head. The six leg-servos allow for flexible leg movements. Three orthogonal servos constitute the 3-DOF hip joint. Two orthogonal servos form the 2-DOF ankle joint. One servo drives the knee joint. The motor distribution is different but the DOF is the same. Table 2 shows the details.

The robot's mechanical sketch could be seen in Fig. 2.

3 Electrical Specifications

This year we compared the advantages and drawbacks over DSP and PC104 and changed our entire electrical architecture to PC104 considering the camera and wireless network. Our electrical controllers include the motor controller and camera controller, specifications of which could be seen in Table. 3. The camera controller works as the main controller processing image identification, location, strategies selection and communications. The movement

Table 2: Motor types and Distributions of DOF

Part	Rotation Axis	Actuator
Neck	Yaw,Pitch	RX-28,RX-28
Shoulder	Roll,Pitch	RX-28,RX-28
Arm	Pitch	RX-28
Hips	Roll,Yaw	RX-28,RX-28
Knee	Pitch,Pitch	RX-28,RX-28
ankle	Pitch,Roll	RX-28,RX-28
	Total DOF	20

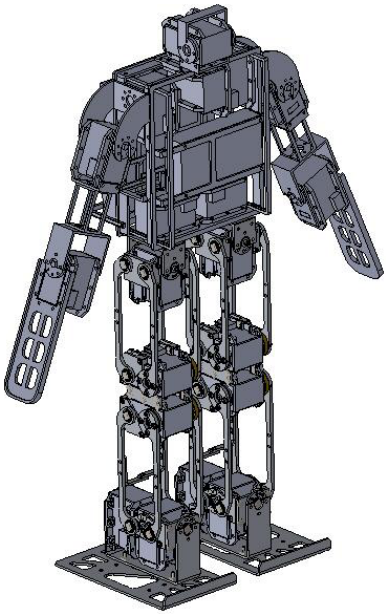

Figure 2: Robot's mechanical sketch

Table 3: Electrical Architecture of our robot

	Camera Controller	Motor Controller
CPU	AMD Geode^{TM}LX800	ATMEL Mega128
FLASH	4GB	128KB
RAM	512MB	64KB
OS	Windows XP Embedded	None
Interface	UART(RS232C)× 2 USB2.0 × 4 Wireless network IEEE802.11g LCD Interface Keyboard and Mouse Interface	UART(RS232 × 1) UART(RS485 × 1) A/D Converter × 8
Sensor	Camera × 1	Attitude Sensor × 1 Including: gyro × 3, Accelerator × 3

and balance maintains are implemented by the motor controller which execute the movement direction from the main controller.

4 Sensors Specifications

There are 4 types of sensors equipped on our robot, image sensors, gyroscopes, accelerators, and potentiometers.

- Image sensor. The camera we choose is Philips SPC 1300 NC. It has a frame rate up to 90 fps and the angle view of 80 degree. This camera is also a very important reason that we changed our electrical architecture from DSP to PC104, since our former camera has a rather bad performance under strong lights. The camera is controller by PC104 and the sample rate is up to 30 frame per second.

- Gyroscopes. Gyroscopes are equipped on the hip of our humanoid robot. It returns the angular velocity for the trunk of humanoid robot and helps to keep the balance of humanoid robot.

- Accelerators. This sensor detects the gravity vector when the robot is static. The main applications of this sensor is that it could be used to recognize whether humanoid robot is standing or lying down. The autonomously getting up from tipping over is depend on this sensor. On the other hand, the dynamic attitude estimate from the fusion of gyros and accelerators is under research.

- Potentiometer. This sensor detects the rotation angle of the actuator. With this sensor, the robot recognizes the current angular position of

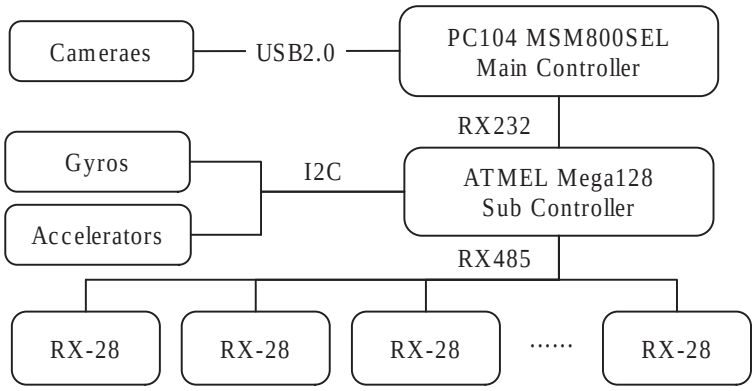

Figure 3: Robot's Electrical Architecture

the joint. This sensor is controlled by actuator controller.

5 Software Specifications

The total electrical architecture could be seen in Fig. 3. So the corresponding software architecture could also be seen in Fig. 4.

Each robot works as an independent agent connected by team messages. The team messages could help to update the world model in each robot. We start our software design from image processing. Segmentation and identification helps us distinguish each objects on the playground. Localization is a complicated fusion of localization of fixed objects and robot locomotion.

6 Conclusion

In this paper, we present the specifications of our robot that has two controllers and 20's DOF. After one year's preparation, ZJUDancer has made a good progress and look forward to the RoboCup 2009. We'd like to share our experience and have a good match between all the teams.

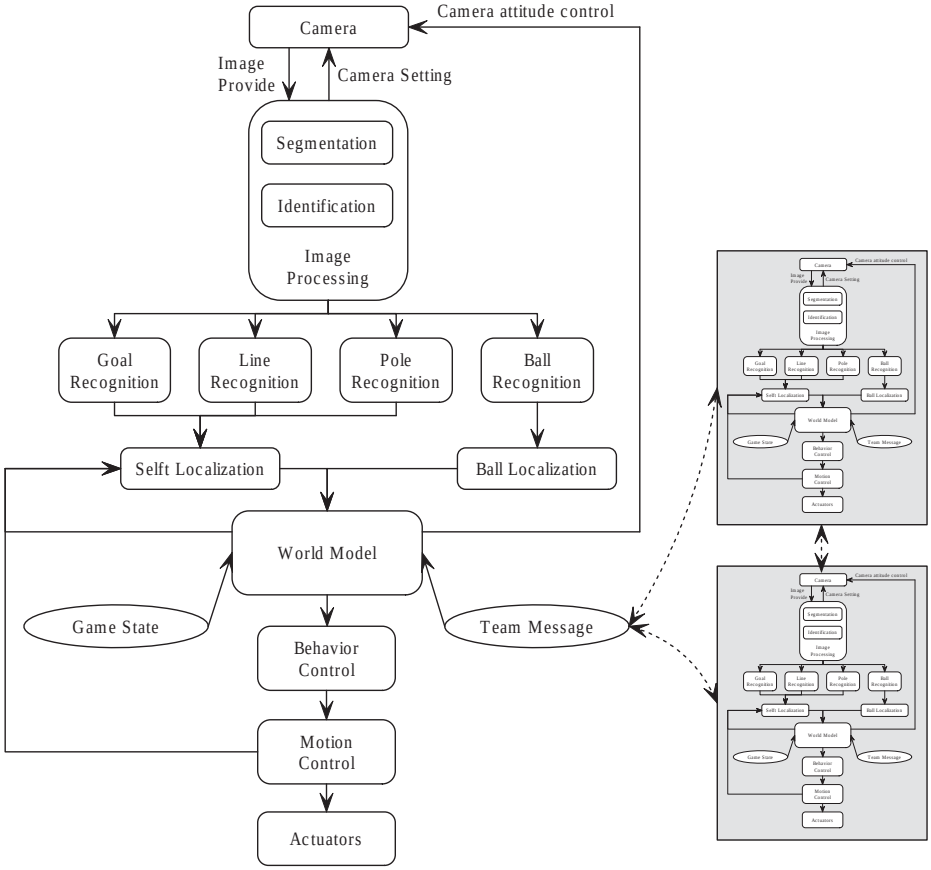

Figure 4: Software Architecture