東北大学機関リポジトリ
TOUR
Tohoku University Repository

Design and Development of a Legged Robot Research Platform JROB-1

著者	近野　敦
journal or publication title	IEEE International Conference on Robotics and Automation, 1998. Proceedings
volume	1998
number	1
page range	146-151
year	1998
URL	http://hdl.handle.net/10097/46651

doi: 10.1109/ROBOT.1998.676343

Proceedings of the 1998 IEEE
International Conference on Robotics & Automation
Leuven, Belgium • May 1998

Design and Development of a Legged Robot Research Platform JROB-1

Satoshi KAGAMI, Mitsutaka KABASAWA, Kei OKADA, Takeshi MATSUKI,
Yoshio MATSUMOTO, Atsushi KONNO, Masayuki INABA, Hirochika INOUE

Department of Mechano-Informatics, The University of Tokyo,
7-3-1 Hongo, Bunkyo-ku, 113-8656 Tokyo, JAPAN

Abstract

A legged robot "JROB-1" is developed for a robotics research platform as a result of inter-university research program on Intelligent Robotics supported by the Ministry of Education Grant-in-Aid for Scientific Research on Priority Areas in Japan. The JROB-1 has the features as follows : 1) self-contained, 2) RT-Linux running on PC/AT processes vision&sensor processing, motion planning and control, 3) connected to a network via radio ethernet as to utilize networked resources, 4) Fujitsu color tracking vision board and Hitachi general purpose vision processing board are attached, 5) all parts are commercially available, 6) it is extensible with respect to sensor, sensor processing hardware and software. JROB-1 is expected to be a common test-bed for experiment and discussion, to do intelligent robotics research by integrating perception and motion.

1 Introduction

A legged robot research platform should satisfy many aspects of research or experiment, from low-level quick/smooth motion to high-level vision/sensor based behavior research in very complex environments. At present, there are many legged robots, but almost all robots are designed for legged locomotion experiments. To satisfy both legged locomotion and high-level behavior by integrating perception and motion, robot should have a good functionality for mechanism, hardware and software.

As a platform, not only functionality, but also self-containedness for free motion, safety for human and robot itself, extensibility for improving its functionality, and easy in conducting experiments, are fundamentally important.

Therefore five key issues are required for a legged robot research platform, a) functionality, b) self-containedness, c) safety, d) extensibility and e) easiness in use. A legged robot "JROB-1" (Fig.1) is developed as a research platform to satisfy these five issues.

Figure 1: JROB-1 Rope Interface Experiment

In this paper, the five key issues required for a legged robot platform is discussed, then design and development of the details of JROB-1 are described. The JROB-1 is a result of inter-university research program on *Intelligent Robotics* supported by the Ministry of Education Grant-in-Aid for Scientific Research on Priority Areas in Japan. Furthermore, rope-interface, vision based target tracking, and voice control experiments using this platform are described.

2 Requirements and Design of a Legged Robot Research Platform

As mentioned above, five key issues are required for a legged robot research platform, those are a) functionality, b) self-containedness, c) safety, d) extensibility and e) easiness in use.

Authorized licensed use limited to: TOHOKU UNIVERSITY. Downloaded on April 12,2010 at 06:48:49 UTC from IEEE Xplore. Restrictions apply.

2.1 Requirements

As for a) functionality, there are three requirements, a-1) body, a-2) computer hardware, a-3) software. A robot body must have enough degrees of freedom and payload. For computer hardware, not only should CPU(s) have enough power and memory, but also vision and other sensor processing peripherals are important. In general, computer hardware consists of a network of heterogeneous processors, but homogeneous or if possible single processor systems are more preferable in developing a robot software simply. For a software, robot software should process sensing/motion decision/control simultaneously. Thus multiple process programming functions are required.

As for b) "self-containedness", wireless is one of the most important features for a mobile robot, from a viewpoint of free motion and free environment. Thus, there are three requirements, that is b-1) power source, b-2) on-board hardware and b-3) wireless network connection capability. Power generator driven by fossil fuels can be used only for a robot outdoors. There are many candidates for on-board hardware, from microprocessor to workstation. Software design is strongly influenced by this hardware design. However in any method, wireless connection to another computer or network is important for remote resource and for software development.

For c) "safety", there are three requirements, c-1) emergency switch for developer, c-2) automatic emergency switch for the robot itself and c-3) independence from software. A robot which has many degrees of freedom, should have safety features for a developer and robot itself, and it should be implemented in mechanism or hardware.

As for d) "extensibility", there are four requirements, that is d-1) easy to attach sensor and sensor processing equipment, d-2) easy to write device driver, d-3) multiple process programming capability, d-4) when heavy use of high level software, the system can use computer resources from outside the body.

To examine many theories, techniques and ideas, it is important to make the robot easy to use. Thus, for e) "easiness", there are two requirements e-1) human interface and e-2) easy to develop software.

2.2 Design

To satisfy a-1), TITAN-VIII and Motor Driver (TITECH MD)[1] are adopted for a body because of the degrees of freedom ($3DOF \times 4$), and of its payload (7kg for each leg). For a-2), PC/AT clone computer, Fujitsu Tracking Vision Board (TRV-CPD6)[2], Hitachi Vision Board (IP2000), and Fujitsu Robot Interface Board (RIF-01)[3] are adopted. For a-3), RT-Linux as for operating system and Euslisp[4] to control the body using 3D geometric model of the body, are adopted.

For b-1), lead-acid battery is adopted to send an electric current to both motor and computer, because of the low resistance inside the battery. For b-2), factory use PC/AT clone board computer which has PICMG connector and backplane which has both ISA/PCI bus are adopted for the on-board hardware. For b-3), radio ethernet equipment (which requires no device driver) is adopted.

To satisfy c-1) and c-2), emergency switch are attached on the top back vertically and also on the both side of front horizontally. For c-3), to make them independent from the software, special emergency circuit is developed, which uses the free/lock function of motor driver, overriding the software output.

To satisfy d-1), RIF-01 interface card are adopted which has 16 channel D/A, 36 channel A/D, 20bits DIO and so on. Also many sensor processing board for PC/AT can be utilized. For d-2), RT-Linux is open-architecture and source code of OS is available, so that a developer can write a device driver for a custom board. For d-3), RT-Linux has multi-process, multi-thread and data transmission method (such as IPC, MSQ). For a hard real-time process, the program should be implemented as a kernel module. For d-4), the system is connected to the network via radio ethernet, so that the system can use NFS, NIS, automount, remote CPU and so on.

To satisfy e-1), rope-interface and voice recognition function are implemented. Furthermore, UHF video signal transmitter is attached to send processing image back to developer for inspecting and debugging. To satisfy e-2), developer can login using radio ethernet, and can inspect/debug/develop the entire software while program is running.

Following, system hardware and software are described in section 3 and 4, and emergency equipment and peripherals are described in section 5 and 6. Furthermore, experiments on JROB-1 are described in section 7.

3 PC/AT Clone Hardware System

3.1 CPU board and Backplane board

CPU board of JROB-1 is PC/AT clone for FA use. It has Pentium 133MHz, 128MBytes EDO-RAM, two serial ports, one parallel port, two IDE connector, floppy disk connector, keyboard connector and SCSI connector. A 810MBytes 2.5inch SCSI harddisk is connected. This board has PICMG connector to connect to a backplane.

Backplane board has one PICMG connector for CPU board, two ISA connector and three PCI connector. Fujitsu tracking vision board (TRV-CPD6), graphics card and network card are attached to PCI bus. Hitachi vision board (IP2000) and Fujitsu robot I/O board are attached to ISA bus (Fig.2).

Authorized licensed use limited to: TOHOKU UNIVERSITY. Downloaded on April 12,2010 at 06:48:49 UTC from IEEE Xplore. Restrictions apply.

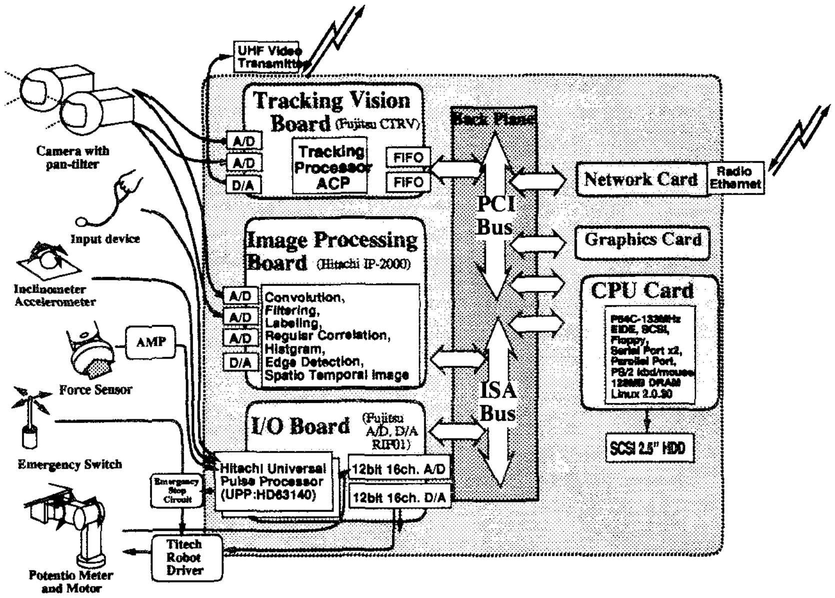

Figure 2: JROB-1 Hardware Components

3.2 Tracking Vision Board (Fujitsu TRV-CPD6)[2]

TRV-CPD6 is a color image processing board which has custom ASIC "ACP" for calculating correlation of a summation of absolute difference between each pixels of two given images. The board has two image inputs and one image output, four image frame memories (one is overlay), and is connected to PCI bus. The board can calculate over 500 correlations between 8×8 reference block and 16×16 search area (Gray scale) at NTSC frame rate (33msec). The board has the features listed below,

- Color correlation calculation,
- Big reference image correlation,
- Masking correlation calculation,
- Distortion array acquisition.

Using this board, tracking, optical flow, finding predefined reference applications are easily available.

3.3 Vision Processing Board (Hitachi IP-2000)

IP2000 is a general purpose image processing board which has custom ASIC for image processing and 16 image frame memory. The board has three image input and one image output, and this board is connected to ISA bus. The board has the features listed below,

- Image operation (logical, arithmetic),
- Deformation (noise reducer, outline, dilation, erosion, shrink),
- Convolution (smoothing, edge emphasis, laplacian),
- Mini-Max Filter,
- Histogram,
- Labeling,
- Regularized Correlation.

These functions are processed at 24MHz (41.7ns/pixel, 11msec for 512×512).

3.4 Robot I/O Board (Fujitsu RIF-01)[3]

RIF-01 is a I/O board which has 16 channel 12bit A/D and D/A, and has two UPP (Hitachi Universal Pulse Processor, HD63140). UPP has 10 channel 10bit A/D, 16 channel DIO connected to UPC (Universal Processor Core) and watch dog timer.

4 Software System on RT-Linux

4.1 Operating System

Real-time processing, there are two methods. One is to use an OS which is especially designed for real-time processing such as Vx-Works, TRON, QNX micro-kernel and OS9, the other is to modify normal OS for real-time processing. Linux is not a real-time OS originally, RT-Linux have two special mechanisms, one is scheduler for real-time process, the other is two-level interrupt handler.

There are many advantages in using Unix operating system as the robot OS, such as,

Authorized licensed use limited to: TOHOKU UNIVERSITY. Downloaded on April 12,2010 at 06:48:49 UTC from IEEE Xplore. Restrictions apply.

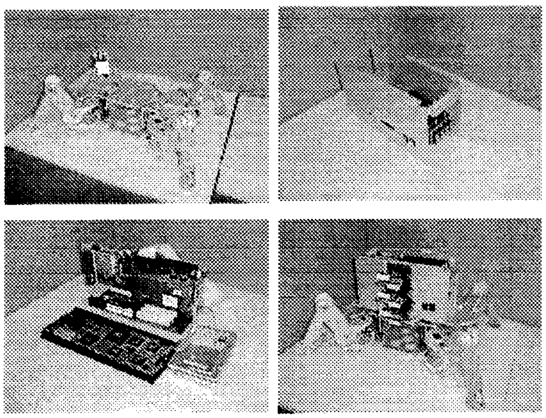

Figure 3: Components, (a) TITAN–VIII, (b) Case for PC, (c) PC parts and (d) After assembled

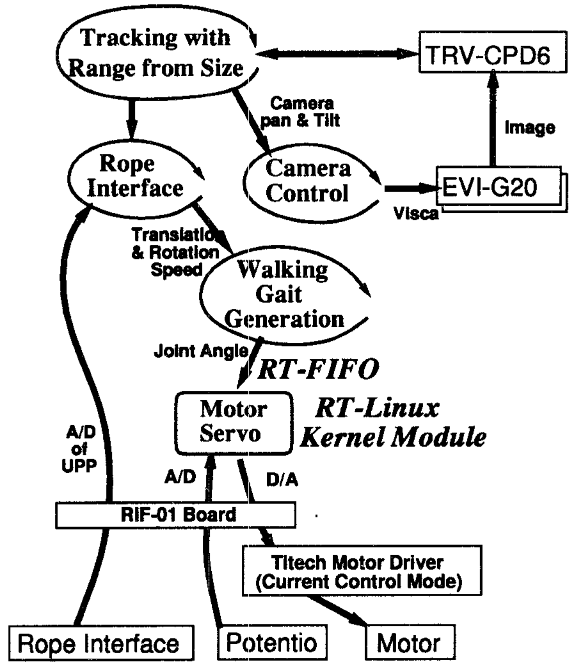

Figure 4: JROB-1 Software Components on Target Tracking Experiment

- Comprehensive develop environment, which has editor, compiler, debugger and GUI
- Parallel programming functions, such as multi-process, multi-thread and inter-process communication
- Network facility, such as NFS, NIS, automount and remote CPU.
- Multi-user.

Especially Linux, there are other advantages, that is

- Source code is open,
- Device driver is easy to write,
- Real-Time package is available, such as RT-Linux, Posix 1.b.

Therefore, as the operating system of JROB-1, we adopted Linux 2.0.30 with RT-Linux package ver.0.5.

Under RT-Linux, user must write a device driver as a kernel module for hard real-time process. User processes communicate with hard real-time process through FIFO.

4.2 Motor Servo

Motor servo for 12DOF actuator is implemented as one RT-Linux hard real-time module (Motor Servo in Fig.4). There are two modes input for this module, one is position of each actuator, and the other is the end point of each leg. In the latter mode, the module calculates inverse kinematics to acquire the current position of each actuator. The module also gets the gain for PD control of each actuator, so that the module reads A/D data of each actuator and decides output force using PD gain and interpolation of designated position, and outputs through D/A. This cycle takes 10msec.

4.3 Walking Gait Generation

Walking gait generation is a user process (Walking Gait Generation of Fig.4). This process has two modes, one is trot walking, and the other is static walking. In trot walking gait, processing cycle is 1.0 $\sim$ 2.0sec. The input of this process is length of translation and angle of rotation of a step, then this process calculates movement of each leg.

4.4 Model based control by Euslisp

Since the robot has many degrees of freedom, it is hard to generate stable body motion in a complex environment. To solve this problem, 3D geometric model is implemented in Euslisp[4] (Fig.5). This special lisp is developed for robot software, and has the features as follows: 1) object oriented, 2) 3D solid modeler, 3) multi-thread programming functions, 4) original X-window toolkit, 5) OpenGL support, 6) efficient memory management using Fibonacci and Mark & Sweep type GC, 7) Unix system-call functions, 8) interface to other programming languages, 9) multiple platforms support (Solaris, SunOS, OSF-1, Irix, Linux, Windows95).

So far, under Remote-Brained approach, robot modeling library has been developed in Euslisp[5]. We applied this result to generate stable walking pattern in various environment.

Authorized licensed use limited to: TOHOKU UNIVERSITY. Downloaded on April 12,2010 at 06:48:49 UTC from IEEE Xplore. Restrictions apply.

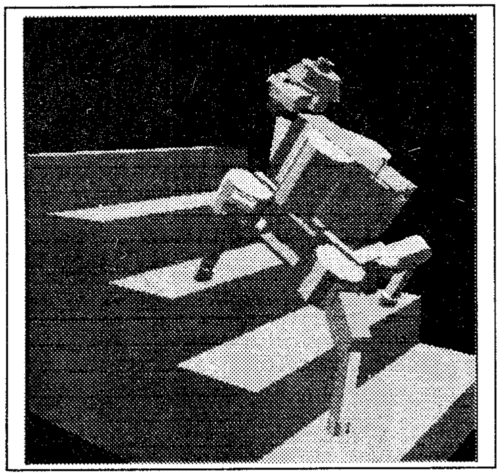

Figure 5: Geometric Model of JROB-1 in Euslisp

5 Emergency Switch and Circuit

Since total weight of JROB-1 is heavy (31.5kg) and motor is powerful, emergency function is important to protect human being and the robot body itself. An emergency switch and two touch switches are attached on the body, and the emergency circuit overrides the output of the computer. When a switch turns on, all motors become free.

6 Peripherals

6.1 Weights and Battery

Weight of TITAN-VIII legs are 18kg, battery is 4kg(two) and total weight of other body parts is about 9.5kg. For battery, two or four lead-acid battery (12V, 4.5Ah, 2kg) are used for both computer and motors, and it can supply power for about 30 or 60 minutes respectively. In trot walking time, maximum current for motors is about 50A and average current is about 4A(96W). Current for computer is average 4.1A(98W). Major power consumption parts are shown in Table 1.

6.2 Controllable Cameras (Sony EVI-G20)

Two controllable cameras including pan-tilter (Sony EVI-G20) are attached on the head. Functions of these cameras (pan, tilt, zooming, iris, focus and so on) are controlled using VISCA protocol through the two serial ports of motherboard. Output image of two cameras are separated to two image processing board so that every board can process both inputs.

Table 1: Power Consumption of Major Parts

Parts Name (number)	Power (W)
Motor Driver(12)	42W
Motherboard include CPU	25W
Color Tracking Vision	8W
Camera with Pan-Tilter(2)	7W
Radio Ethernet	4W
Graphics Card	3W

6.3 Force Sensor on Foot (FSR)

Four force sensing resisters are attached on the front legs, so that the force and force direction of leg end can be measured. Each data are amplified and are measured by A/D of UPP of RIF-01 board.

6.4 Inclinometer and Accelerometer

Inclinometer and Accelerometer are attached on the center of the body, and each data are measured by A/D of UPP of RIF-01 board.

7 Experiments

Three experiments, rope interface, voice interface and vision based tracking behavior are described. Rope interface and voice interface are a good way for make experiment easy. Vision based tracking behavior shows an example of correlation technique using on board Fujitsu tracking vision.

7.1 Rope Interface Experiment

Rope Interface is an intuitive interface for human being. Basically, robot moves toward the direction that rope is pulled. Amount of rotation and translation is calculated from that direction. When pulling from the front, robot moves mainly translation along to the front direction. But when pulling from the back, robot mainly rotates to turn to that direction.

Fig.1 shows the Rope Interface experiment outdoors carrying about 5kg of fruit. Software components are shown in lower side of Fig.4, circle arrow shows a user process on RT-Linux, rectangular box shows hardware, rounded box shows RT-Linux kernel module.

7.2 Voice Interface Experiment

Voice interface is one of the easiest ways to control the robot. Since voice recognition requires a lot of computer resource in general, we use workstations on the network through wireless LAN. Spectrum Subtraction and Speech recognition are done by commercial software HTK (Box area in Fig.6).

Authorized licensed use limited to: TOHOKU UNIVERSITY. Downloaded on April 12,2010 at 06:48:49 UTC from IEEE Xplore. Restrictions apply.

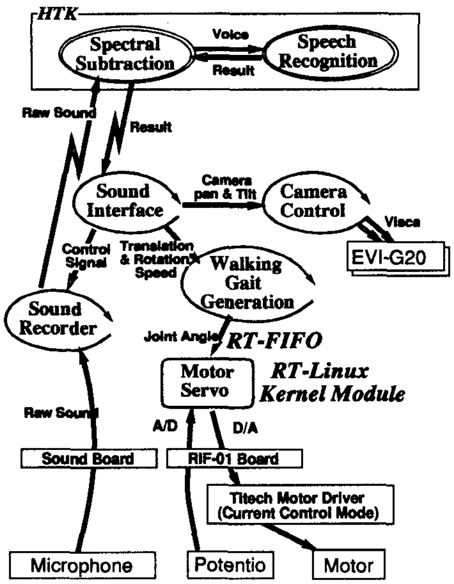

Figure 6: Software Components of Voice Interface Experiment

7.3 Vision based Tracking Behavior

We combined transformation ratio with subsampling ratio and developed a logarithmic algorithm to get accurate range information from the size of the target in single view[6].

The system generates a linear series (h_i) of N templates depicting the object at different sizes from $H_0/2$ (half its original size), to H_0. This series is generated by simply scaling the original image using affine transformation (Digital Differential Analyzer).

Features smaller than $H_0/2$ and larger than H_0 can be matched using *exactly the same template series*, but scaled up or down as appropriate. In general, if a feature can be matched to template h_i of size S, its range R and maximum error ΔR are determined as:

$$R = \frac{2D_0S_0N}{(N+i)S}, \quad \Delta R = \frac{D_0S_0}{2NS}. \qquad (1)$$

Within the limits of the precision of the experiment, range can be quite accurately calculated.

Fig.4 shows software components of target tracking experiment, circle arrow shows a user process on RT-Linux, rectangular box shows hardware, rounded box shows RT-Linux kernel module. "Tracking with Range from Size" module tracks the given target and try to move the body to keep the target in front center and constant distance.

8 Conclusion

In this paper, the five key issues, "a) functionality, b) self-containedness, c) safety, d) extensibility and e) easiness", required for a legged robot platform is discussed. Then the design and development of the details of JROB-1 are described.

A legged robot "JROB-1" is developed as for a research platform to satisfy these five issues. The JROB-1 is a result of inter-university research program on *Intelligent Robotics* supported by the Ministry of Education. Four equipments, a four legged robot TITAN-VIII[7], Fujitsu Tracking Vision Board (TRV-CPD6)[2], Fujitsu Robot Interface Board (RIF-01)[3] and Motor Driver (TITECH MD)[1] are also developed through this program on intelligent robotics. These four components are combined and PC/AT clone computer controls entire body by RT-Linux including 12DOF legs, Tracking Vision, Hitachi Vision Processing Board (IP2000), cameras, sensors, rope interface and so on in real-time. Emergency stop sensor and circuit are attached to keep both human being and robot safe. Total system costs about 24,000 US$.

Furthermore, three experiments using this robot (rope interface, voice interface, vision based tracking behavior) are described.

Acknowledgement

This research has been supported by Grant-in-Aid for Scientific Research on Priority Areas of the Ministry of Education, Science and Culture of Japan (No.07245102), and by Grant-in-Aid for "Research for the Future Program" of the Japan Society for the Promotion of Science (JSPS-RFTF96P00801).

References

[1] F. E. Fukushima, T. Tsumaki, and S. Hirose. Development of a PWM DC Motor Servo Driver Circuit (in Japanese). In *Proc. of 13th Annual Conference of Robotics Society of Japan*, Vol. 3, pp. 1153–1154, 1995.

[2] T. Morita, N. Sawasaki, T. Uchiyama, and M. Sato. Color Tracking Vision System. In *Proc. of 14th Annual Conference of Robotics Society of Japan*, Vol. 1, pp. 279–280, 1996.

[3] Y. Matsumoto, K. Sakai, T. Inamura, M. Inaba, and H. Inoue. PC-based Hypermachine: A kernel System for Inteligent Robot Application. In *Proc. of 15th Annual Conference of Robotics Society of Japan*, Vol. 3, pp. 979–980, 1997.

[4] T. Matsui and S. Sekiguchi. Design and Implementation of Parallel EusLisp Using Multithread (in Japanese). *Information Processing Society*, Vol. 36, No. 8, pp. 1885–1896, 1995.

[5] M. Inaba. Remote-Brained Robotics: Interfacing AI with Real World Behaviors. In *Robotics Research: The Sixth International Symposium*, pp. 335–344. International Foundation for Robotics Research, 1993.

[6] Alan Lipton and Satoshi Kagami. Real-World Path Following for a Monocular Vision based Autonomous Mobile Robot in a 'Remote-Brain' Environment. In *Seventh Australian Joint Conference on Artificial Intelligence (AI'94)*, pp. 490–497, 1994.

[7] K. Arikawa and S. Hirose. Development of Quadruped Walking Robot TITAN-VIII. In *Proc. of International Conference on Intelligent Robots and Systems (IROS'96)*, Vol. 1, pp. 208–214, 1996.

Authorized licensed use limited to: TOHOKU UNIVERSITY. Downloaded on April 12,2010 at 06:48:49 UTC from IEEE Xplore. Restrictions apply.