The Fourteenth International Symposium on Artificial Life and Robotics 2009 (AROB 14th '09),
B-Con Plaza, Beppu, Oita, Japan, February 5 - 7, 2009

Quadruped Virtual Robot Simulation in Virtual Environment Obeying Physical Law

Kouki MIYOSHI, Norihiro ABE, Yoshihiro TABUCHI
Kyusyu Institute of Technology
680-4 Kawazu, iizuka, Fukuoka 820-8502, Japan
Email:miyosi@sein.mse.kyutech.ac.jp

Hirokazu TAKI
Wakayama University
930 Sakaedani, Wakayama-shi
Wakayama 680-8510, Japan
Email: taki@sys.wakayama-u.ac.jp

Shoujie He
VuCOMP
Richardson, Texas, USA
Email: hesj@computer.org

***Abstract*:** In the development of a robot, the validation of a robot with use of real machinery takes much cost and time. Especially it is difficult to validate a robot's behavior at the unsafe place. Then developers have paid attention to virtual debugging. Using a program validated in VR space makes verification of a real machine's behavior more efficient.

In this research, we have a virtual robot walk on a road autonomously by using the program that the virtual robot tracks a line on virtual environment.

Keywords: *VR, Virtual Robots, Virtual Space*

1. INTRODUCTION

In the last few years, many robots are advancing to human society with the development of robot industry, and a robot is expected to be a member of the general public in the near future. However the validation of robot motion control with using real machinery takes much cost and time to develop a real robot and no one deploys any robots in dangerous environment for fear of damage owing to violent fall or collision. If robots damage in real space, it will compel us to pay much time and expense for fixing robots. This will increase in necessary expense and length of a period for robot development.

Then virtual debugging using virtual reality space has gotten attention. Virtual debug enables developers to validate the robot motion in dangerous environment in parallel with development of a real machine after deciding the specification and design of a robot. A debug of a control program using a real machine debug will be shortened using the control program that successfully operates in virtual debug.

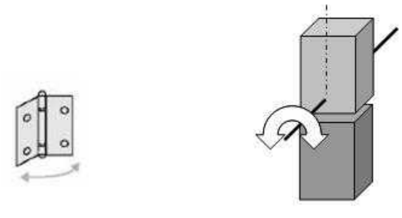
In this research, we have a virtual four legged robot built in virtual space based on physics move autonomo usly in a dangerous environment. By analyzing images c aptured with two cameras installed on the four legged ro bot, we have it track a line on the environment including bridges and up down-hills and walk on a road whose edges are white lines.

2. Construction of Virtual Reality Space

In the simulation, it is necessary to construct th e virtual reality space based on a physical rule bec ause of making virtual environment same as real on e as far as possible. So we use the rigid physics c alculation library, Vortex (developed by CMLabs Si mulations, Inc.) to build the virtual reality space. V ortex has the function to create a basic object like plane, box, corn, sphere, and cylinder. We enable to give a constant restraint between objects by using joint. We are able to create a composite object wit h the use of two or more basic objects. Combinatio n of composite objects and joint enables to express a complex object like robot and car.

3. Expression of Virtual Robot

In this research, servomotors of virtual quadruped robot are expressed with a hinge joint between two rectangular solids and two boxes as shown in Fig.1.



(a) Hinge joint (b) Servomotor
Fig.1. A Virtual Servomotor

A virtual quadruped robot is composed by 13 composite objects and 12 hinges concatenating them. Shoulders, upper legs, and lower legs consist of boxes and cylinders. These parts are linked with a hinge joint. An appearance of robot is shown in Fig.2.

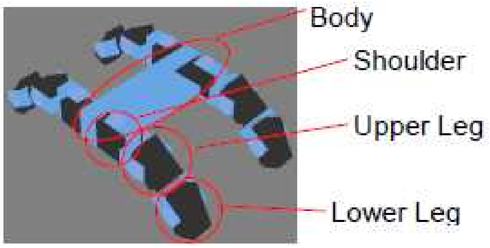


Fig.2. The Quadruped Robot

4. Virtual Sensor

The robot has virtual camera. System gets imag e buffer from virtual camera to control the robot. However images captured with cameras will decline (s hown as Fig.4), when a robot goes ahead. So i n addition to cameras, a virtual gradient sensor is inst alled on the robot to calculate how much cameras decline and compensate the image to make an imag e processing easy.

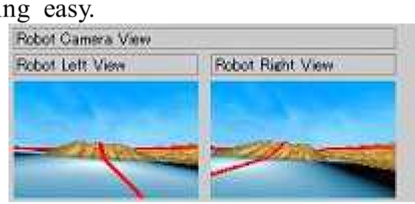


Fig.4. Images from two virtual cameras

A virtual gradient sensor obtains the gradient an gle of camera from two coordinates (x_1, y_1) and (x_2, y_2) as shown in Fig.5.

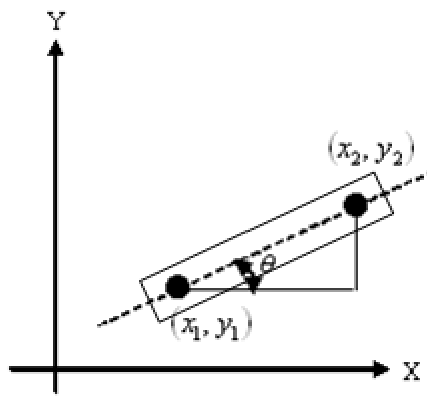


Fig.5. The Virtual Gradient Sensor

The gradient θ is calculated with the use of th e expression (1).

$$\theta = \sin^{-1}\left(\frac{x_1 - x_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}}\right) \qquad (1)$$

5. Image Processing

5.1. Autonomous walk along the line

There is the case where a single continuous line in the original image is cut into several ones after rotation. So System expands the line before rotatio n. Then the robot tries to find a line to track, but several lines are detected in the images. As it seem s that the line to be tracked exists near the robot, i nitial points to find the corresponding line from eac h camera are determined based on the proximity to the robot.

To make a walking command, the left image is used as a base image. The first step is to discover a red point from the left image. The second step is to find from the right image the point corresponding to the red one by extending the horizontal line passing through the y coordinate value of the red point in the left image as shown in Fig.6. This process is run over until a red point is found in the both images. The average number of x-coordinate of the right and the left images decides that the robot is on line or deviates from side to side.

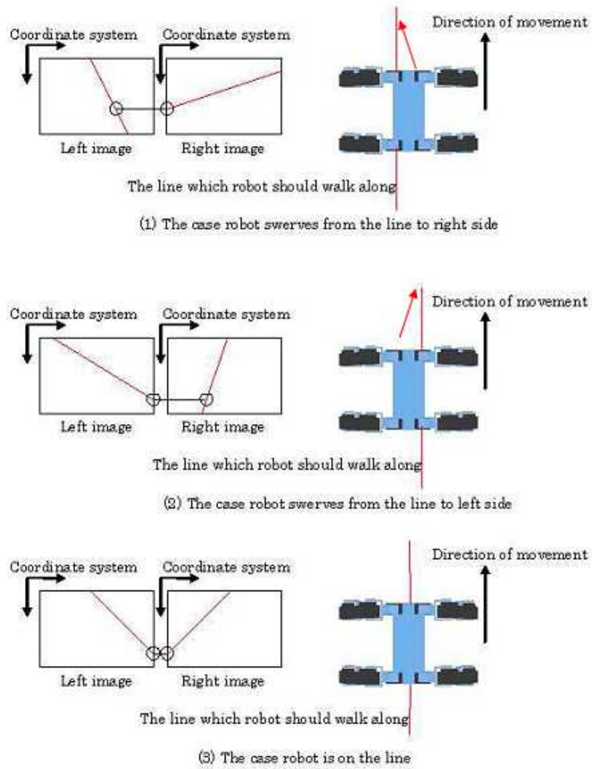


Fig.6. The Decision of a Walk Order

If the robot is on the line, the average x-coordinate is equal to the half size of the image width. If the robot deviates to the right side, the average x-coordinate will be less than the half size of the image width. On the other hand, in case the robot deviates to the left side, the x-coordinate will be more than the half size of the image width (Fig.6). Even if a robot is going straight, as it will shake from side to side, the x-coordinate dose not precisely coincide to the half size of the image width. So the decision of whether the robot is on the line or not is relaxed. If a robot is judged to be on the left side, it will be given a command to move to the right direction, and vice versa.

5.2. Autonomous walk along the road

We make a robot walk along the center line between the left and right white lines. A center line finder looks for cross points where a line parallel to the horizontal line intersects with left or right white lines by moving the line from the bottom of left and right images. If the left and right cross points are found, then the mid point is on the centerline. If only one cross point is found, the center line is extended parallel to the white line found from the end point of the centerline found immediately before. If no cross point is found, as it means a robot to be in a junction, the robot has only to keep going straight until a new white line is found.

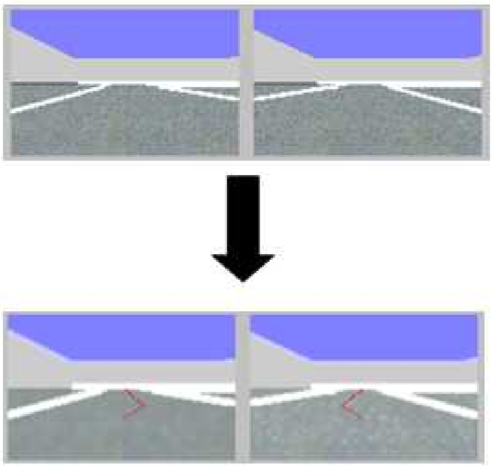


Fig.7.Plot the Imaginary Center Line

6. Conclusions and Future Work

This research's aim is to make a virtual robot autonomously walk in the Virtual reality space and we successfully implement simulation. Now the robot enables to walk autonomously along the line and the road. In the future, we want a robot to walk in the environment with diverging roads. Given the start point, goal point and a map, we expect a robot to go automatically from the start to the goal.

Now, except for up hills or down hills, the system successfully simulates the behavior of a robot on a flat plane with a constant homogeneous friction coefficient. Next we would like to simulate the behavior on rolled ground with variable friction coefficients.

7. Acknowledgment

We greatly appreciate the aid of the Grant-in-Aid for Scientific Research (S) and (A).

8. Reference

[1] Norihiro Abe, Yoshinori Takamura, Kazuaki Tanaka, Hirokazu Taki (2003): Construction Of Physics Simulation Environment for Education Using Vortex, In *proceedings of the SCI2003*, 301- 307.

[2] Toshiaki Oomori, Norihiro Abe, Kazuaki Tanaka, Hirokazu TAKI, Shoujie HE (2005): Concurrent development of Virtual Robots and Real Robots Based on Physical Low, *IMEKO / IFAC / IFIP Workshop on Advanced Robot Systems and Virtual Reality. ISMCR2005.*

[3] Toshiaki Oomori, Norihiro Abe, Hiroaki Taki, Shoujie He (2007): A System Allowing Concurrent Design and Implementation of Both Virtual and Real Robots , accepted, *International Symposium on Artificial Life and Robotics (AROB 2007).*

[4] Yi PAN, Norihiro Abe, AND Kazuaki Tanaka (2003): The Virtual Debugging System for Embedded Software Development, *SCI2003*, 78-84.

[5] Yi Pan, Norihiro ABE, Kazuaki Tanaka (2003): The Virtual Debugging System for Embedded Software Development, *VRAI2003.*

[6] Yi Pan, Norihiro Abe, Kazuaki Tanaka, Hirokazu Taki (2004): The Virtual Debugging System for Developing Embedded Software Using Virtual Machinery, In *proceedings. of Embedded and Ubiquitous Computing, International Conference (EUC 2004),* Aizu Japan, 85-95.