Internet Coordinated Pet Robot Simulator based on MSRDS

Byoung-Wook Choi, Sung-Hun Lee and Soo-Yeong Yi
Department of EE, Seoul National Univ. of Tech., Seoul, Korea
(Tel:+82-2-970-6412; E-mail: bwchoi@snut.ac.kr, dash@snut.ac.kr, suylee@snut.ac.kr)

Abstract: This paper proposes a pet robot simulator (PRS), based on MSRDS, which supports interactively controlled two walking robots connected over network. To be pet robot simulator, modeling a commercial pet robot is performed and gait planning is also implemented. By using MSRDS software platform, we connect two robots which are displayed together but controlled separately over network. The two walking pet robots can be simulated interactively by joysticks. It seems to be an internet game for pet robots.

Keywords: Robot simulator, Pet robot, Network control, MSRDS, Internet game.

1. INTRODUCTION

In general, software platform including development environments and services can be determined as an effort that guarantees the productivity and reliability of robots through the reuse of software.

The well-known commercial products of robot software platform are ERSP (Evolution Robotics Software Platform) by ER (Evolution Robotics) and MSRDS (Microsoft describes Robotics Developer Studio) by Microsoft. The ERSP has been largely used in the implementation of application components, and the MSRDS that is a developmental environment including dynamic simulator has been extended to laboratories at universities and conventional MS tools-based users [1, 2].

The MSRDS provides the DSS (Decentralized Software Services) that is a service-based runtime architecture and distributed application pattern, which is a type of REST (Representational State Transfer), and CCR (Concurrency and Coordination Runtime) that is concurrent processing and control technology in distributed computing environment. In addition, it provides a dynamic simulation engine using the PhysX [3].

In recent years, a graphic simulator for security robots using the MSRDS software platform and dynamic simulation environment [4] and a service-based simulation framework using environmental information were presented [5]. But, the most simulators using the MSRDS are applied to mobile robots [4, 6].

This paper presents a PRS (Pet Robot Simulator) which is a graphic simulator coordinated over network and is aimed to be a useful tool for planning gaits or dynamic behaviors for walking pet robots. We performed gait control for two virtual robots coordinated over network with considering their dynamic movement according to the speed of its movement using a joystick. In addition, we developed a GUI for applying it to control the gait of an actual pet robot. The two walking pet robots can be simulated interactively by joysticks. It seems to be an internet game for pet robots.

2. SYSTEM ARCHITECTURE

Fig. 1 illustrates the implementation of two pet robots in the PRS, which are controlled by different users A and B connected over the network. Users A and B operate the pet robots implemented in each computer using GUI. While the connection is maintained as a simultaneous manner, each user applies same commands to both robot A and B in A’s and B’s computers. The robots are controlled by the joystick and the direction values presented in the joystick will be applied to move them to left, right, forward and backward directions according to the walking algorithm.

3. GRAPHIC MODEL OF PSR

Although a certain graphic simulator can perform 3D modeling like OpenGL or 3DMAX, a dynamic simulation function that simulates not only simple kinematical movements but also collision and collapse between joints caused by the unstable walking is required. This study developed a dynamic simulation environment for pet robots using the PhysX, which is a dynamic simulation engine implemented in the MSRDS. A dynamic model for certain objects can be presented by variables employed in the objects, such as weight, elastic force, gravity in a simulation environment, and friction. Similarly, it is possible to perform a dynamic walking in robot movements in which the feet of the robot contact the floor and push it using the frictional force between the feet and the floor. Thus, a developer can develop and verify various gaits of the walking robot in the PRS before developing a practical pet robot.

All objects moved in a virtual space can be classified as independent objects. Thus, in the design of the exterior parts of a robot using a 3D design program, each part is to be generated as a format of file for each object. The quadruped walking robot used in this study has four legs, and each leg...
has shoulder, upper leg, lower leg, and foot. Thus, this study modeled a total of 19 objects, such as 16 components in four legs including body, neck, and head. Fig. 2 shows the graphic model of a pet robot and displays the kinematical information of the four-legged walking pet robot. All joints were established based on a cylindrical graphic model and the head and feet were designed based on a hexagonal model. Texture information was applied using the actual data of the robot modeled by using the Pro-E. In addition, the data will be applied to the surface of the robot by transforming it into MSRDS objective files [7]. The robot has three degrees-of-freedom per leg and consists of the links for the two operational axes for moving head and neck. Thus, a total of 14 degrees-of-freedom are applied to the robot.

4. RESULTS

The movement of a four-legged robot in the PRS can be performed by two ways. First, the commands to move the body of the robot are applied to the determined direction directly, and the commands to move legs are applied to repeat the forward movement only. This method can be applied to a static simulation. Second, the movement of the robot is performed by the foot in which the foot pushes the floor, and the body can be moved by the friction between the foot and the floor [8]. It is a process occurred in the real world in the movement of animals and can be used in the case of the simulation, which can be implemented by a dynamic simulation. The dynamic simulation should support a physical environment that includes various natural laws, such as gravity, friction, elastic force, contact between objects, and collision. Fig. 3 illustrates the falling down of the robot that is bumped against the camera installed at the front during forward movement. Thus, it showed that the robot in a virtual space is not fixed in a point and stands on the floor based on the gravity by supporting four legs.

In the results of the stable gait walking order implemented in this study and the control of the robot, the robot showed stable walking in a dynamic simulation space [9]. This study determined an input of a forwarding signal using a mouse through a direction controller or a pushing of a joystick to forwarding movement for moving each leg of the robot for one step. Fig. 4 shows the forward movement of the robot. As shown in Fig. 4, the picture 1 shows the initial pose before moving, and the other sequential pictures represent the walking based on the mentioned walking algorithm.

As the same as the forward movement the left and right turning can be performed by the left and right direction signals from the direction controller of the Dashboard through the dragging of a mouse or controlling of a joystick. The movement of the turning is similar to the forward movement except for its direction. Fig. 5 represents the left turning of the robot by 180°.
V. Conclusions

This study developed the PRS which is a simulation environment for a four-legged robot based on the MSRDS environment. The MSRDS that applies a physical engine corresponding physical environments, which are difficult to deal in the existing static simulation program, such as OpenGL, generates some virtual environments through establishing lots of natural phenomena as specific data as similar as that of the real world. It is useful to determine dynamic behaviors in the legged robot walking in real environments.

In these problems, the first problem is difficult to estimate the accurate movement as the robot contacts an object because the external physical body of the robot is not designed using actual texture mesh information. In the MSRDS models, it can be occurred at not external surfaces but the contact by the modeled object [2].

Second, the problem can be caused by the commands of composed entities instead of applying actual motor operation values for the rotational torque in each joint for the walking of the simulator. As a result, there are some difficulties in the precise study of gaits between the movement limitation and the robot walking. Thus, it is necessary to develop an extra dynamic modeling method for the frictional coefficient between the floor and the foot of the robot and its torque.

Finally, the problem can be caused by the application of commands over a network that should be applied to both the robot implemented in the user A's computer and the robot implemented in the user B's computer at the same time. In the case of the commands that are to be applied at the same time, as the user A accesses the user B's computer to control the A's robot, although the user A can control the A's own robot in the B's computer, it is difficult to control the robot in the A's computer. Because there are some unstable conditions in the network and time delay even though the same commands are transferred through the network, the error occurred in the robots between two computers gradually increased according to the passage of time. Therefore, it is necessary to solve the problem of the synchronization between two systems in which the commands are not to be applied to both the A's and B's computers simultaneously but the way that the control commands are only applied to the A's own computer, and the movement information of the robot is to be applied to the robot implemented in the B's computer through the network. In addition, a study on the time delay occurred in the distributed environment is required.

It can be considered that the pet robot simulator, PRS, implemented in this study can be usefully applied to investigate remote brains and gait, and demonstration of the robot.

REFERENCES