Information Exchange for Controlling Internet Robots

Soon Hyuk Hong, Ji-Hwan Park, Key Ho Kwon, and Jae Wook Jeon

School of Information and Communication Engineering, Sungkyunkwan University
Chunchun-Dong, Jangan-Gu, Suwon City, Korea 440-746
jwjeon@yurim.skku.ac.kr

Abstract. In order to control a remote robot over the internet, the operator needs to be able to exchange information with it in real-time. However, the camera image generally used for providing feedback information is too large to send in real-time over the internet. Furthermore, it takes a long time for the operator to send complex robot commands repetitively. These time delays in exchanging feedback and command information between the remote robot and its operator make it difficult to control internet robots efficiently. This paper proposes an information exchange technique for controlling a remote robot over the internet, in which the desired information is extracted from the image in order to reduce the time required to send the feedback information, and a task-level command is used to reduce the time required to send the robot commands. As a result, the proposed technique can help an operator to control a remote robot in real-time over the internet.

Keywords: internet robot, robot, information exchange.

1 Introduction

In the early stages of robotics, remote robots were controlled through a dedicated network, however, more recently, remote robots have been controlled over the internet, because of its use of a standard communication protocol and the fact that it is available almost everywhere. Remote robots controlled over the internet are called internet-based robots or internet robots. The first internet robots to be introduced were stationary remote robots operating in a limited geographical region [1-2] and, subsequently, autonomous internet robots operating in a wider region were developed [3-9].

It takes a long time to transmit large amounts of information over the internet because of its narrow bandwidth and unpredictable transmission time. Therefore, in order to control internet robots, abstract commands that do not contain detailed information about the task at hand and reduced feedback data should be used.

A virtual teaching environment was used as a teaching interface for assembly tasks in [10]. In the virtual teaching environment, the teaching motion of the operator is formulated into symbolic operation commands, which are sent to an actual robot in the form of task commands. Instead of using detailed information to control a robot, symbolic task-level commands such as “Grasp part A” are sent. In the autonomous mobile robot called Xavier described in [4], several computer systems were used for control, perception, and planning. If the operator selects a block in the two-dimensional building map

through a web-base interface, Xavier can visit the corresponding desired office using its autonomous navigation system. In [5], an overhead camera and an additional camera in the mobile robot supplied images to the local operator. The local operator could then provide the mobile robot with movement commands by pointing at a particular location in these images. The coordinates of the designated location were then sent to the mobile robot.

In the case of remote robots, in addition to the robot commands, the feedback information is also very important. Consequently, various data exchange techniques have been developed for the purpose of reducing the transmission time of this feedback information. In [11], fourteen magnetic sensors and one range sensor were used to track the human motion, and their data were sent to the host computer in order to render the human motion in 3D. Instead of sending all of the data, which consisted of nine double floating point numbers per sensor, a quaternion consisting of four double floating point numbers per sensor was sent to the host computer, in order to reduce the data transfer time and to improve the graphic rendering speed. In [12], the data size can be changed according to the priority, which is determined by the data content and the task condition. If the control information is more important than the image information, then the image data is divided into several parts and each part is transmitted with the whole set of control data through the network.

This paper proposes an information exchange technique for controlling a robot over the internet, which used a task-level command to reduce the time required to send commands to the robot and extracts the desired information from an image in order to reduce the time required to send the feedback information. The proposed information exchange technique is applied to a mobile robot that can be operated by either manual control or supervisory control over the internet. The manual control mode requires fast information exchange and is therefore different from the supervisory control mode.

In section 2, an internet robot system is described. In section 3, the information exchange technique for controlling this internet robot is proposed. In section 4, the operator interface of the internet robot is explained in detail. Some experiments are described in section 5. Finally, our conclusion is given in section 6.

2 An Internet Robot System

As shown in Fig. 1, a local operator can control the mobile remote robot over the internet using either the manual or supervisory control mechanisms. The local operator can use the web-based robot interface to control the remote robot. In order to send information about the tasks that result from the local operator commands, a web server at the remote site sends either camera images of the remote robot or the locations of the remote robot calculated from these images. The remote robot receives the commands from the operator over the internet and performs the desired tasks.

If the operator gives a command to the remote robot, the remote robot performs its corresponding task and the remote web server returns the result of the task to the operator. In this way, the operator can follow the status of the robot and its tasks and formulate subsequent commands based on the information provided by the web server.

3 The Information Exchange Technique

In order to improve the real-time performance in a communication network, various data exchange techniques have been developed with the goal of reducing the communication time [1, 2, 5]. We propose an efficient technique for exchanging information between the operator and the remote robot for both the manual and supervisory control modes, as shown in Fig. 1.

The proposed technique simplifies the robot commands in order to transfer them to the remote robot quickly. Since it does not take a long time to transfer these simple commands, the local operator can easily control the remote robot over the internet.

For the manual control mode, the proposed technique uses eight keyboard inputs for the robot commands employed to make the remote robot move. These keyboard inputs can adjust the velocity and direction of the remote robot, as shown in Table 1. Using these keyboard inputs, the operator can easily drive the remote robot in the “hands-on” control fashion.

For the Supervisory control mode, the proposed technique uses the mouse to select a destination point or via-points designed to make the remote robot move and issue the corresponding command messages in the form of a sequence of short messages to the remote robot.

In order to transfer the information sent by the remote robot, the proposed technique uses two kinds of feedback data, the image sequence and the robot’s location on the captured image. In order to transmit the real image sequences, RTP (Realtime Transport Protocol) provided by JMF (Java Media Framework) is used, and the robot location information is transmitted by peer-to-peer socket communication.

Fig. 1. An internet robot system

The robot’s location can be obtained from the image sequence in real-time, while the camera is capturing the successive images. Since it may take a long time to transfer the image sequence over the internet, the robot’s location can be used as complementary

information to the image sequence. As shown in Fig. 2, while the camera at the remote site is capturing the images, the robot's location can be obtained using an elliptical tracking algorithm that can track the moving object as the primitive, an ellipse, by calculating the gradient of the contour of the ellipse [13]. Then, both the robot's location and the image sequence can be transmitted over the internet. The information pertaining to the robot's location can be transmitted to the local site faster than the image sequence information, because its size is smaller than that of the image. Thus, the operator is informed of the robot's location and can control the remote robot in a timely manner, even though the image data has not yet been received.

4 Operator Interface

We developed an operator interface to allow the operator to control the remote robot easily. In order to construct the GUI, a Java applet is embedded in the web page and the JMF (Java Media Framework) is used to provide the image of the remote robot. The web server provides the GUI and the operator can access the web page used to download it in the form of a Java applet. Then, the operator can make the TCP/IP connection to the remote robot directly for the purpose of controlling the robot. The operator can monitor the robot's movement through the live images obtained from the web server.

In order to provide the operator with the necessary visual information, we use one camera and the JMF toolkit. In particular, RTP (Realtime Transport Protocol) is used to achieve better performance for the peer-to-peer connection. As shown in Fig. 3, the operator interface consists of a Virtual workspace, Vision image viewer, State message board, and Robot control panel.

The Virtual workspace displays the robot's current position. Also, the operator can select a position in the virtual workspace, in order to move the robot along the desired path and so be able to visualize the robot's movement in the simulation before sending the corresponding commands to the real robot.

The Vision image viewer shows the image of the camera connected to the web server at the remote worksite. The real image also provides the operator with important feedback information in the manual control mode, even though the images are transmitted with some delay.

The State message board shows the information of the operator interface and the communication results from the robot. The information contained in the State message board facilitates the task of the operator.

Table 1. Commands for a remote robot

Number	Acceleration	Deceleration	Description
8 (↑)	a	d	Forward movement
2 (↓)	a	d	Backward movement
4 (←)	a	d	Left turn
6 (→)	a	d	Right turn
7	a	d	Forward left turn with a radius
9	a	d	Forward right turn with a radius
1	a	d	Backward left turn with a radius
3	a	d	Backward right turn with a radius

Fig. 2. Transmission of information about the remote robot

Fig. 3. Operator interface

The operator interface provides the operator with three control modes. In the Direct Control Mode, the operator can exert manual control over the robot and, thus, can control the robot's movement by using the keyboard. The operator determines the robot's position through the Vision image viewer and controls the robot by pressing the keyboard so as to adjust the robot's velocity and direction. The other two control modes are the Supervisory Control Mode and the Task Scheduling Mode. In these modes, the operator can use the mouse to indicate the subsequent via-points or a destination point. The 'Select position' button in the operator interface is used for this purpose. In the Supervisory Control Mode, the operator can make the robot move by indicating the next position and pressing the 'Move one-step' button. Then, by checking the robot's movement in the corresponding image, the operator can determine the robot's subsequent movement and continue to operate the robot. The Supervisory Control Mode is a basic control mode for Internet robot control, because it provides a very simple interface which allows the operator to move the robot in a visual fashion [1, 2, 4]. In the Task Scheduling Mode, the operator can predict the robot's movement

by indicating a set of via-points corresponding to the path to the destination and previewing the resultant robot movement by means of a simulation. There is a ‘Set position’ button used for storing the via-points and a ‘Simulation run’ button for simulating the robot movement in the virtual workspace.

Although many Internet robots have been designed which use standard communication protocols to make the connection and a simple user interface, these robots have some limitations. When a large number of users access these systems, the web server has to make a new process for each request, thereby increasing its overhead. Therefore, it is not possible to achieve an adequate level of interactivity using this method [1, 2]. In the proposed technique, the use of the Java platform allows the user interface to be more interactive and the performance to be enhanced with the standard communication protocol.

Fig. 4. Manual control of the remote robot

5 Experiments

In the system shown in Fig. 1, we performed manual control with the Internet based mobile robot by using short command messages and feedback information.

The operator can use the operator interface described in section 4 to control the robot. In the operator interface, the operator can select the 'Use Keyboard' check button to perform manual control using the eight directional keyboard inputs. In this way, the operator can easily drive the remote robot in the "hands-on" control fashion.

The image sequences that provide the feedback information are susceptible to be transmitted over the internet with some delay, depending on the environmental conditions. Even though the local and remote sites were near to each other, the image sequence sometimes froze or successive frames were not synchronized and, consequently, it was necessary to replay part of the sequence. However, the additional information sent in the form of short messages was successfully received with only a small delay and overlaid as the ellipse on the real images that represents the robot's current position. This small delay comes from the capturing of the raw image and processing of the tracking algorithm. Although the original algorithm was slightly modified before applying it to our system, it showed good performance in the tracking of the moving robot.

Fig. 4 shows the manual control of the remote robot, where the robot's location is represented by an elliptical tracker. In this mode, the additional information provided by the image sequences is useful in controlling the robot remotely over the internet.

6 Conclusion

This paper proposes an information exchange technique for controlling an internet robot, which makes use of a task-level command for the purpose of reducing the time required to send robot commands and extracts the desired information from an image in order to reduce the time required to send the feedback information. We also developed an operator interface that uses the proposed information exchange technique for controlling the internet robot easily. Using the developed operator interface, the internet robot can easily be driven by means of the manual and supervisory control modes.

Acknowledgements. This research was supported by the MIC, Korea under ITRC, IITA-2006-(C1090-0603-0046).

References

1. Ken Goldberg, Michael Mascha, Steve Gentner, Nick Rothenberg, Carl Sutter, and Jeff Wiegley. : Desktop Teleoperation via the World Wide Web. IEEE International Conference on Robotics & Automation. (1995) 654-659
2. Eric Paulos, John Canny. : Delivering Real Reality to the World Wide Web via Telerobotics. IEEE International Conference on Robotics & Automation. (1996) 1694-1699.
3. Paul G. Bakes, Kam S. Tso, and Gregory K. Tharp. : Mars Pathfinder Mission Internet-Based Operations Using WITS. IEEE International Conference on Robotics & Automation. (1998) 284-291

4. Reid Simmons, Joaquin L. Fernandez, Richard Goodwin, Sven Koenig, and Joseph O'sullivan. : Lessons Learned from Xavier. *IEEE Robotics & Automation Magazine*. June (2000) 33-39
5. Patrick Saucy, Francesco Mondada. : KhepOnTheWeb: Open Access to a Mobile Robot on the Internet. *IEEE Robotics & Automation Magazine*. March (2000) 41-47
6. Dirk Schulz, Wolfram Burgard, Dieter Fox, Sebastian Thrun, and Armin B. Cremers. : Web Interfaces for Mobile Robots in Public Places. *IEEE Robotics & Automation Magazine*. March (2000) 48-56
7. Sebastien Grange, Terrence Fong and Charles Baur. : Effective Vehicle Teleoperation on the World Wide Web. *Proceedings of the 2000 IEEE International Conference on Robotics & Automation*. April (2000) 2007-2012
8. Kuk-Hyun Han, Shin Kim, Yong-Jae Kim, Seung-Eun Lee and Jong-Hwan Kim. : Implementation of Internet-Based Personal Robot with Internet Control Architecture. *IEEE International Conference on Robotics & Automation*. May (2001) 217-222
9. Paul E. Rybski, Ian Burt, Tom Dahlin, Maria Gini, Dean F. Hougen, Donald G. Krantz, Florent Nageotte, Nikolas Papanikolopoulos, and Sascha A. Stoeter. : System Architecture for Versatile Autonomous and Teleoperated Control of Multiple Miniature Robots. *IEEE International Conference on Robotics & Automation*. May (2001) 2917-2922
10. Hiroyuki Ogata, Tomoichi Takahashi. : Robotic Assembly Operation Teaching in a Virtual Environment. *IEEE Transactions of Robotics and Automation*, vol 10, no. 3. (1994) 391-399
11. Tom Molet, Ronan Boulic, Serge Rezzonico, and Daniel Thalmann. : Architecture for Immersive Evaluation of Complex Human Tasks. *IEEE Transactions on Robotics and Automation*, vol. 15, no. 3. June (1999) 475-485
12. Takafumi Matsumaru, Syun'ichi Kawabata, Tetsuo Kotoku, Nobuto Matsuhira, Kiyoshi Komoriya, Kazuo Tanie, and Kunikatsu Takase. : Task-based Data Exchange for Remote Operation System through a Communication Network. *Proceedings of the 1999 IEEE International Conference on Robotics and Automation*. May (1999) 557-564
13. Stan Birchfield. : Elliptical Head Tracking Using Intensity Gradients and Color Histograms. *IEEE Conference on Computer Vision and Pattern Recognition*. June (1998) 232-237