Multi-touch Interface for Controlling Multiple Mobile Robots

Jun Kato

The University of Tokyo

School of Science,

Dept. of Information Science

jun.kato@acm.org

Daisuke Sakamoto

The University of Tokyo

Graduate School of Information

Science and Technology

d.sakamoto@gmail.com

Masahiko Inami

Keio University

Graduate School of Media Design

inami@designinterface.jp

Takeo Igarashi

The University of Tokyo

Graduate School of Information

Science and Technology

takeo@acm.org

Abstract

We must give some form of a command to robots in order to have the robots do a complex task. An initial instruction is required even if they do their tasks autonomously. We therefore need interfaces for the operation and teaching of robots. Natural languages, joysticks, and other pointing devices are currently used for this purpose. These interfaces, however, have difficulty in operating multiple robots simultaneously. We developed a multi-touch interface with a top-down view from a ceiling camera for controlling multiple mobile robots. The user specifies a vector field followed by all robots on the view. This paper describes the user interface and its implementation, and future work of the project.

Keywords

multi-touch interface, multiple-robot operation, human robot interaction, home robot, entertainment robot

ACM Classification Keywords

H5.2. Information interfaces and presentation (e.g., HCI): User Interfaces – Interaction styles; I.2.9. ARTIFICIAL INTELLIGENCE: Robotics – Commercial robots and application

1. Introduction

All robots, including those that do their tasks autonomously, do not work without an instruction by

users. We therefore need interfaces for giving instructions to them.

Single robots are normally controlled with joysticks, keyboards, and other pointing devices. However, with advances in robotics, variations of user interfaces for these purposes have become wider. For example, multimodal interfaces such as combination of hand gestures and speech for one assistant robot [1], and a portable interface using personal digital assistant (PDA) for mobile robots [2] have been proposed. This system allows users to navigate a robot with waypoints projected on the screen. Recently, user studies of intuitive interfaces [3] have been performed, in which motor impaired users have a robot grab things with laser pointers and a touch-screen with buttons on it. Among these studies, it has been concluded that the invention of advanced user interfaces helps operating single robots.

Handling tasks with multiple robots is also desirable, because they can do various tasks with greater efficiency than a single robot. However, multiple robots substantially increase amount of information exchange with their users who have to maintain situational awareness and continue operation. It often makes the manner of operation complex and difficult. Users have a limitation in the capability of their attention, so they cannot see too much information displayed either at the same time or time-multiplexed. Therefore, many user interfaces for operating single robots do not work effectively. Designing interactions between people and multiple robots to achieve their effective cooperation has been a difficult research issue.

Upon the cooperation of people and multiple robots, existing studies have tried to give robots some intelligence. This approach aims to allow for limited resources of people to care the situation and make orders continuously. Some have continued studies which only tell robots initial state and make them work autonomously [4]. Others like Fong et al. insisted that completely autonomous approaches are not yet feasible, and that robots should engage in dialogue with their users when required [5]. Their study indicates two factors for effective cooperation of people and robots. The first factor is that roles and responsibilities against tasks are clearly separated between users and robots. It is generally said that users should have responsibility for global tactics, and robots for local tasks. When the distinction between global and local tasks is unclear, problems occur for both the robot and the user. The second factor is that users can command robots easily as possible. Here we need richness of user interfaces cultivated in the field of HCI. Discussion of Driewer et al. describes how the user interface in human-robot teams should be [6]. They pointed out that in teams consisting of people, robots and their supervisor, the use of graphical user interfaces (GUI) greatly affect the performance of their tasks.

In this paper, we propose an intuitive interface using a multi-touch display to control multiple mobile robots simultaneously. Users get a top-down view from a ceiling camera in real time, which is virtually overlaid with a 2-dimensional vector field. All robots follow the vector field. Users can manipulate this field by touching and passing their hands on the display. Therefore, they can easily control all robots through the multi-touch display.

2. Multi-touch Control Interface
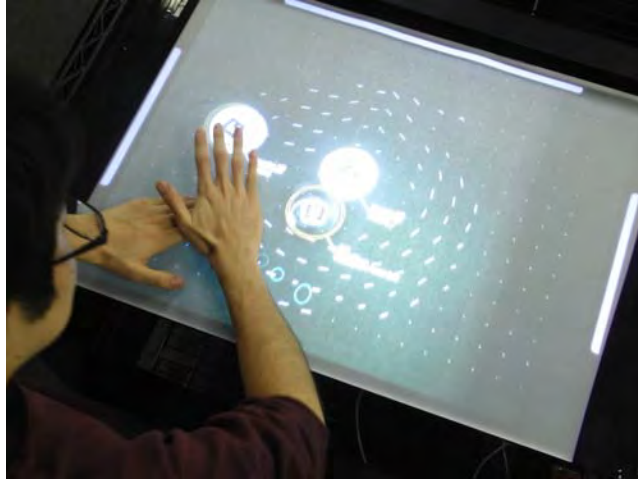
Figure 1. Users touch and pass their hands on the panel to control multiple robots.

We developed our interface so that users can easily maintain their situational awareness and control movements of multiple robots intuitively. Robots are expected neither roles nor responsibilities more than their function of moving to some place where users intended. That is to say, roles for robots and users are clearly defined. Through the interaction design described below, we aim to satisfy the key factors for effective cooperation of people and robots.

2.1. Top-down view
We decided to use a tabletop panel with a top-down view from a ceil-mounted camera projected on it. When we use our interface for home use or entertainment use, in most cases such as using robots in a room, physical position of the environment is static. We might capture images of top-down view with cameras with such an

environment. Even if setting cameras on the ceiling is difficult, it is possible to construct a map of top-down view gradually with the feedback information of sensors of robots. Our main point is using a top-down view with which users can see the global situation at once.

2.2. Focus on the Field
Although users can maintain their situational awareness with the top-down view, they have to switch their attention among all robots when they command robots individually. Designing the interactions between users and robots as users treat robots collectively can help bring down attention switches. We think focusing on the field working as a proxy of all robots instead of individual robots may work.

Virtual Force Field [10] proposed in the dawn of collision avoidance studies used a global potential field to decide where robots should go locally. Each observed obstacles have some potential, in other words, areas around obstacles are positioned virtually high and difference in the height brings robots to a lower place. Regardless of obstacles, the potential field can be used to guide robots. A user interface that we can increase or decrease potential of the position we push [11] was developed and used for entertainment purposes such as video games. This interface can be used to operate real robots. But we think pushing the display is not an enough interaction in order to make users feel like moving robots along their arms. It produces little move in horizontal plane, while robots mainly move horizontally.

Instead, we use the motion of touched surface on the panel. We make grids in the field to hold 2-dimensional vector information. When touched surface moves on
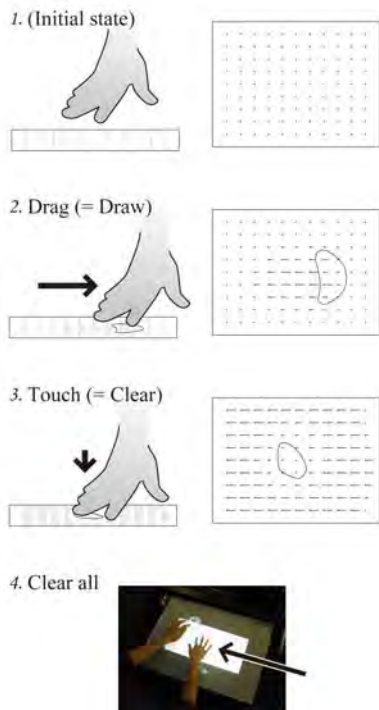
Figure 2. Vector fields under typical operations available in our system
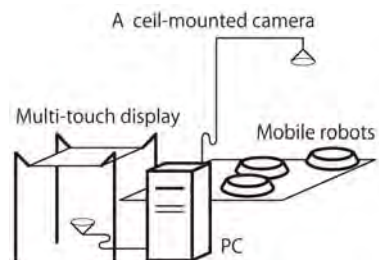
Figure 3. System overview

the panel, grids in and near its track are affected to remember the direction of the motion. All robots decide in which direction to go according to the sum of vectors of grids near their position.

2.3. User Interface

The vector field starts to hold its vector data when users pass their hands on it, but all the vectors shorten over time. In a figurative sense, all streams made by hands get thinner as time goes on.

Operations currently available in our system are shown below.

- Drag – When users touch and pass their hands on the panel, a virtual stream appears on the vector field and robots move according to the stream.

- Touch – Touching the panel without motion will clear vectors under the surface. Thus, we can stop a robot by touching an area where we predict it may go through.

- Clear all – There is a panel including a button labeled "clear the vector field" off the screen. To access this panel, there is a handle at the right edge of the panel. By grabbing the handle and pulling it to the center of the screen, we can use the button to stop all robots at once.

- Mix – When users drag their fingers on existing streams, vector data near the streams are blended in proportion to distances against streams. Nearer streams affect stronger on the vectors.

3. Implementation

3.1. Hardware

Our multi-touch interface adopts the low-cost method proposed by Han et al. [9], using frustrated total internal reflection of infrared light in the acrylic panel. The shape of the touching surfaces can be detected by infrared camera set under the panel. We set a downward-pointing camera on the ceiling. We used Roomba and Create robots, both by the iRobot Corporation[1].

3.2. Software

Our system is built using the Java platform, and checked the software properly works on Mac OSX and Windows. Bluetooth links the robots to the computer with JSR-82 implementation for connection. Images from the ceiling-mounted camera are captured at 1/30 fps through QuickTime in Mac OSX or DirectShow in Windows. Positions of the robots are calculated through detecting markers in the captured images using ARToolKit[2]. In our implementation, captured images work as both intuitive interface and a sensor. Shapes of luminous areas on the multi-touch display are visually detected using a marching square algorithm; and through approximation and calibration, we get position and size information of touched surfaces. All surfaces are approximated by ellipses whose shape can be expressed by using the center position and two lengths of major and minor axis.

We divide the screen into a defined set of grids, and hold 2-dimensional vector data in each grid. In our environment, the grid interval is 46 pixels, which is

---

[1] http://www.irobot.com/

[2] http://www.hitl.washington.edu/artoolkit/

actually 15 cm long on the floor. Every time the information of the luminous areas is updated, their motion is tracked using a very simplified algorithm for optical flow. We simply recognize the nearest luminous area in the previous image as the same, continuously moving area as usual. When the distance of current area and the nearest in the previous image is further than a threshold value, however, it is not recognized as the same but the new area. Tracked motion affects the existent vector field in the manner that grids directly under the touched surface are completely overwritten with the motion vector, and that those near the surface are blended with it in proportion to the distance against the center of the surface. Grids further than a defined distance (92 pixels i.e. 30 cm) are not affected. All vector data is shortened to a defined rate (98%) every time the information from the camera is updated. So, after neglected for a while, the vector field will convergent to the initial state.

Every robot moves to the direction calculated by adding vector data of grids near it. This addition uses a reverse manner as a luminous area affects grids near it. That is, nearer grids affect strongly and further ones weakly in proportion to distance. Grids outside a certain circle whose center is the position of robot are neglected. In our current implementation, robots can only rotate or move forward. When the difference between the calculated and current direction is above a defined threshold (±10 degrees), the robot starts to rotate instead of going forward.

4. Discussion
We have not taken a formal user test, so we recognize need for it. On that basis, we describe our point of view we acquired after the private test in the laboratory.

4.1. Top-down View
Our interface can track locations of robots globally. In our future work, we may record and play the waypoints they passed. We can play a record without affecting real robots, where we can implement time seeking by choosing a visited waypoint instead of clicking a certain point on a normal seek bar. Applications for sweeping robots may allow users to register some of their favorite actions like cleaning only around a desk, avoid a trash-box, and so on, and play them whenever needed. Furthermore, for entertainment use, a supervisor can see the field from the god view to make multiple robots interact with audiences in real time.

4.2. Focus on the Field
Our interface allows users to do task with robots intuitively, and in some cases, it delivers unique movement of robots that cannot be achieved by other operating methods. Meanwhile, there found to be equally some limitations. For example, our current implementation can make robots go around a static loop like a circle for many times, by drawing a stream with its beginning and the end connected. This cannot be achieved by teaching them waypoints with pointing devices. On the other hand, the vector field cannot bring robots on a path going across itself. In addition, it is difficult to operate only one robot despite the others.

Inspired by the user interface using a hand-drawn sketch to control robots [9], we may adopt a method which allows users to draw a virtual wall that cannot be crossed by robots. Erasing tool and menu buttons for changing modes should also be equipped. Along with this approach, we may aim to solve the limitations by adopting other user interfaces as exceptional operations on the basis of the vector field manipulation.

Another possible approach is to define virtual objects on the field with positive potential which alienate robots. Users can drag and drop them working as if they were sheep dogs chasing mobile robots as sheep. This approach is similar to ours in that it makes users focus on a few virtual things without dividing capability of attention into each robot. Here we may combine the concept of boids [11]. It defines movement of robots bound by simple equations about their relative position. When users chase robots with a virtual sheep dog, robots as boids with a proper relational equation may succeed in escape without collision among themselves.

5. Conclusion

We developed a multi-touch interface to control multiple mobile robots simultaneously by manipulating a vector field on a top-down view from a ceiling camera. Our study implies that enhanced HCI can offer a partial solution for the bottleneck of current HRI problem such that people have limited capability of attention. We are going to extend our implementation to achieve more enhanced usability and accomplish more complex task using multiple robots.

6. Acknowledgements

7. References

[1]   Rogalla, O., Ehrenmann, M., Zollner, R., Becher, R., and Dillmann, R. Using Gesture and Speech Control for Command a Robot Assistant. *In Proc. IEEE International Workshop on Robot and Human Interactive Communication*, IEEE Press (2002), 454-459.

[2]   Fong, T., Thorpe, C., and Glass B. PdaDriver: A Handheld System for Remote Driving. *In Proc. IEEE International Concefence on Advanced Robotics*, 2003.

[3]   Choi, Y.S., Anderson C.D., Glass J.D., and Kemp, C.C. Laser pointers and a touch screen: Intuitive Interfaces for Autonomous Mobile Manipulation for the Motor Impaired. *In Proc. ACCESS Conference on Computers and Accessibility*, ACM Press (2008), 225-232.

[4]   Pham, D.T., Awadalla, M.H., and Eldukhri, E.E. Adaptive and cooperative mobile robots. *In Proc. IMechE*, 221 (3), 279-293, 2007.

[5]   Fong, T., Thorpe, C., and Baur C. Multi-Robot Remote Driving With Collaborative Control. *In IEEE Trans. on Industrial Electronics*, 50(4), 699-704, August 2003.

[6]   Driewer, F., Sauer, M., and Schilling, K. Discussion of Challenges for User Interfaces in Human-Robot Teams. *In Proc. European Conference on Mobile Robots*, 2007.

[7]   Han, J.Y. Low-cost multi-touch sensing through frustrated total internal reflection. *In Proc. UIST 2005*, ACM Press(2005), 115-118.

[8]   Borenstein, J., and Koren Y. Real-Time Obstacle Avoidance for Fast Mobile Robots. *In IEEE Trans. on Systems, Man and Cybernetics*, 19(5), 1179-1187, October 1989.

[9]   Fukuchi, K., Rekimoto, J. Marble Market: Bimanual Interactive Game with a Body Shape Sensor. *In Proc. ICEC 2007*, Springer (2007), 374-380.

[10] Skubic, M., Anderson, D., Blisard, S., Perzanowski, D., and Schultz, A. Using a hand-drawn sketch to control a team of robots. *Autonomous Robots*, 22(4), 399-410, May 2007.

[11] Craig, W.R., Flocks, Herds, and Schools: A Distributed Behavioral Model. *Computer Graphics*, 21(4), 25-34, July 1987.