Gesture-Based Interface for Connection and Control of Multi-device in a Tabletop Display Environment

Hyunglae Lee[1], Heeseok Jeong[2], Joongho Lee[3], Ki-Won Yeom[3], and Ji-Hyung Park[3]

[1] Department of Mechanical Engineering, Massachusetts Institute of Technology, USA
hyunglae@mit.edu
[2] Game Development Center, NHN Corporation, Korea
heesuk0@naver.com
[3] Intelligence and Interaction Research Center, Korea Institute of Science and Technology, Korea
{yap153,pragman,jhpark}@kist.re.kr

Abstract. In this paper, we propose a gesture-based interface for connection and control of multiple devices in a ubiquitous computing environment. With simple selection and pointing gestures, users can easily control connections between multiple devices as well as manage information or data between them. Based on a robust gesture recognition algorithm and a virtual network computing technology, we implemented this concept in an intelligent meeting room consisting of a tabletop, interactive wall displays, and other popular devices such as laptops and mobile devices. We also performed a preliminary user study in this environment, and the results show the usability of the suggested interface.

Keywords: Gesture-based interface, multi-device connection.

1 Introduction

In today's ubiquitous computing environment, we are exposed to various types of computing devices such as desktops, laptops, and mobile devices. While dealing with these devices, any user might have had the experience of wasting time in managing connections and controlling the device. Solving the connection problems in using multiple computing devices, ranging from the above-mentioned ones to emerging devices such as tabletops and interactive surfaces, has become an important issue in HCI research.

In this paper, we propose a novel gesture-based interface to eliminate the cumbersome processes in connection and control of multiple devices in a tabletop and interactive surface environment. With simple selecting and pointing hand gestures, users can easily connect the tabletop with the surrounding wall screens. In addition, this interface enables users to connect their personal devices, ranging from a laptop to a mobile device, with public ones, such as wall screens or a tabletop display. After connection has been established, users can control the connected device using the controlling device or transmit data by selecting the data and pointing to the connected device.

J.A. Jacko (Ed.): Human-Computer Interaction, Part II, HCII 2009, LNCS 5611, pp. 216–225, 2009.

A robust hand and fingertip detection algorithm and a virtual network computing technology are integrated to realize this concept. We implemented the suggested interface in an intelligent meeting room consisting of a tabletop, three interactive wall displays, several laptops, and mobile devices. A preliminary user study in this environment showed that the suggested interface enables users to easily manage connections between multiple devices, and to intuitively handle data.

2 Related Works

There have been several studies on virtual connections of multiple devices in a tabletop and interactive surface environment. Here, a virtual connection means a network connection between physical devices which allows users to control one device while using another one as if two are physically connected. There are several researches [19, 24] and commercial software supporting virtual connections [5, 17, 25], even though most of them are specially designed for remote PC connections.

i-Land [23] extends the collaboration environment to public devices including a tabletop to wall screens. However, it only supports the transportation of whole documents from the tabletop to wall displays. *Table-centric space* system [7] enables moving objects from the tabletop to the wall screens by introducing the concept of "World in Miniature (WIM)." MultiSpace [13] is a similar example consisting of a tabletop display and wall screens. The user can send the data to the wall screen or laptop by dragging it onto an appropriate portal located in the corner of the tabletop display. Although the above two systems provide new interfaces between the tabletop and wall screens, users might feel inconveniences in matching correspondences between several WIMs or portals and multiple screens. *i-Room* [3] is another example supporting virtual connections of multiple devices for efficient collaborations. Although *PointRight* [4] in this environment allows a single mouse and keyboard to control multiple screens, there still exists the inconvenience in interfacing devices.

Over the last decade, there have been many researches done on the virtual connecting of multiple devices through natural and intuitive means as well as researches on the direct manipulation of remote information. Rekimoto proposed a pen-based interface called *Pick-and-Drop* [22] which allows a user to pick up an object on one device and drop it on the other one as if the two were physically connected. *Slurp* [9] is similar to the *Pick-and-Drop* interface except it uses an eyedropper instead of a pen. *That one there* [6] interface uses tags and a gesturepen for data sharing. A user can transmit data by pointing the gesturepen to the device with a tag. *Sync Tap* [20] and *Touch-and-Connect* [26] enables virtual connections by pressing a button synchronously and touching the device, respectively. A virtually connected medium called *tranStick* [1] was also suggested that functions as a virtual wire. Users can easily establish, change, and close connections between devices using a pair of *tranSticks* with the same identifier. *Gaze-Link* [2] is another intuitive means of virtual connection in which a user can establish a virtual connection over a network by looking at the target through a camera on his/her computer. Rekimoto also proposed a new interface called *FEEL* [21] by introducing a nearfield communication channel such as RFID or infrared communication technology. A user can establish a wireless connection by bringing two devices closer together.

Although there are several advantages in using physical media, closeness, touch, and gaze information, users might feel uncomfortable whenever moving to near the devices to establish network connections. In addition, there exist inconveniences in carrying the devices. On the other hand, in our approach, users need not move near to the device or carry any special device in their hands for network connections. We formerly suggested a new interface concept called Select-and-Point in [8]. In this paper, we fully implemented the concept and extended the Select-and-Point interface to multiple devices, including laptops and mobile devices, in the tabletop and interactive wall displays environment.

3 System Description

The suggested interface provides users with a simple and intuitive interface and interaction style in connecting and controlling multiple devices through a simple selecting and pointing gesture. Users can easily establish, change, and close network connections between devices by selecting the device of interest and pointing to the other one. After a connection has been established, users can control the connected device using their own device as if the two are physically connected. In addition, users can easily transmit data by pointing to the connected device after selecting the data to send. We implemented the interface successfully by combining a virtual network computing technology and a robust hand-gesture recognition algorithm.

3.1 System Configuration

The system consists of three parts: a presence server, a controlling peer, and a controlled peer. The architecture of the system is described in Fig. 1.

The presence server consists of a connection management module, a gesture recognition module, and a networking module. The connection management module takes charge in handling the network connections that enables virtual network connections between multiple devices based on the Internet Protocol Address, port number, and ID number of each device. The gesture recognition module analyzes vision information from cameras to interpret users' hand gestures. The networking module is commonly used in the presence server, controlling peer, and controlled peer to transmit or receive data between them.

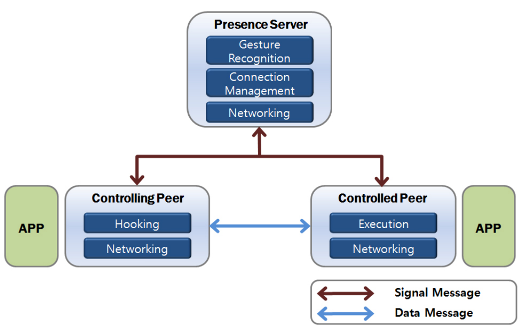

Fig. 1. System architecture of the suggested interface

The controlling peer is installed on the controlling device and the controlled peer on the device to be controlled by the controlling peer. For example, when a user controls the interactive wall screen using his/her laptop, the controlling peer and controlled peer is embedded in the laptop and the wall screen computer, respectively. The hooking module in the controlling peer detects users' input events such as keyboard inputs, mouse events, or file selections. The execution module in the controlled peer generates input events such as keyboard or mouse, and executes the file transferred from the controlling peer.

A protocol consists of signal messages and data messages. Signal messages are messages related to connections and disconnections between the presence server, controlling peer, and controlled peer. Data messages are transmitted between the controlling peer and controlled peer, and carry information to operate the connected device. Message types used in the system are described in Table 1.

Table 1. Message types used in the protocol

Type	Message Description
Registration/ Termination	Register a new controlled peer to a presence server or terminate existing registration
Control-ready	Ready message from a controlling peer to a presence server to control the selected device
Selection	Message from a controlled peer to a presence server to notify the device selection
Address	Message from a presence server to a controlling peer to inform IP address and port number of the device to be connected
Connection Complete/ Termination	Message from a controlling peer to a presence server to inform the completion / termination of network connection
Keyboard	Message from a controlling peer to a controlled peer to transmit keyboard events
Mouse	Message from a controlling peer to a controlled peer to transmit mouse events
File	Message from a controlling peer to a controlled peer to transmit the selected file information

Network connections are independent from the type of device and each device can be either controlling or controlled (see Fig. 2). In general, personal devices ranging from laptops, to PDAs, to mobile phones are connected with shared devices such as interactive wall screens or electronic boards. The system also supports simultaneous connections. For example, three keyboards can be simultaneously connected to one display to be controlled.

The entire network is based on asynchronous communication to minimize blocking during data transmission.

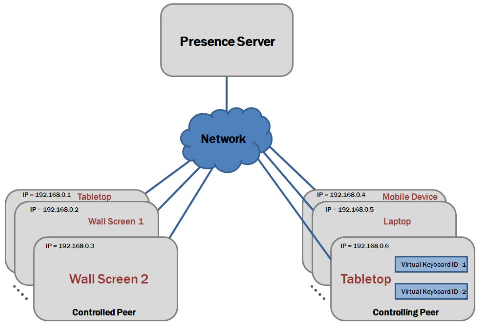

Fig. 2. Support of multiple device connections

3.2 Bare-Hand Gesture Recognition

The bare-hand gesture recognition is essential to implement the suggested interface. Here, bare-hand means that no devices are attached to the user and the user uses his/her hands as an input device. First, we introduced an adaptive background subtraction method to extract the foreground from the dynamically changing background. From the separated foreground, we extract the skin areas by using color information in RGB and HSV color space. Considering that each hand region has a certain size, we eliminate skin pixels that do not contain small circular area. All filtered skin areas are then grouped into several clusters. Then, we check the movement of each cluster by calculating optical flows around each cluster. Clusters whose image velocity is higher than the threshold value are considered as a hand candidate. Among hand candidates, we finally extract hand areas with simple geometric filters that characterize the hand. While scanning the contour of the circle around the center point of each hand candidate with radius bigger than hand size, exactly two intersection points should be encountered or no intersection points exist. Fingertips can be also easily detected from the positions of the hands with a circular shape filter similar with the one used in hand detection. We can easily calculate the direction a user is pointing based on the detected position of hand and fingertip, and also recognize the user's command gestures with Hidden Markov Model algorithm. We used a USB webcam or a network camera in implementation, and the suggested algorithm runs at a full frame rate (30Hz) that satisfies real-time interactions. Details of these concepts, as well as each operation of the hand and fingertip detection algorithm, will be described in a future publication.

4 Implementation in a Tabletop Display Environment

The suggested interface is implemented in an intelligent meeting room called "Intelligent and Responsive Space (IRS) [10, 11]." The IRS, which mainly consists of a tabletop display and three surrounding wall screens, is set up for efficient collaborative works such as team meetings or seminar presentations. Laptops and other mobile devices can be used in this environment. In addition, we installed a pan-tilt-zoom IP camera at the center of the ceiling of this environment to recognize users' hand gestures.

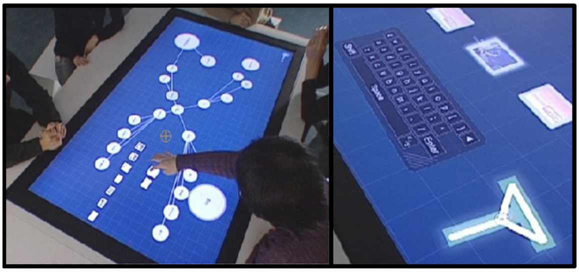

Fig. 3. The tabletop system in the IRS (Left: tabletop display; right: virtual keyboard and "T" icon)

Our tabletop is built as a horizontal table with a 55 inch LCD unit and an infrared LED touch panel supporting high-resolution images of 1920 x 1080. Users can manipulate this system with the fingers. This system also provides command icons called "interactive callout icons" and a virtual keyboard for the ease and diversification of information handling. Users can create command icons and control the system by drawing the shape of icons, each of which has a specific meaning to represent the user's intention. To implement the suggested interface in this environment, we introduced an T-shaped icon which means "ready for a new virtual network connection." The tabletop system, virtual keyboard, and an example of interactive call out icons are shown in Fig.3.

The surrounding wall screens were installed to extend the information sharing space and support effective collaboration.

The suggested interface enables users to easily connect and control devices of interest between the tabletop, wall screens, laptops, and mobile devices. Several examples of multiple device connection and control are given below.

First, a user can connect the tabletop with one of the wall screens through a simple selecting and pointing gesture. After creating the T-shaped icon, the user attaches this icon to the virtual keyboard to be used. This is the selecting process in the tabletop. With a pointing gesture, the indicated wall screen and the virtual keyboard with mouse touchpad are connected over the network. Visual and sound feedbacks notified the success of the connection (see upper right of Fig.4), and the user is able to control the contents of the wall screen with the virtual keyboard on the tabletop as if these two devices were physically connected. The user can also send the file on the tabletop display to the wall screen just by selecting the file of interest and pointing at the wall screen. With this operation, the file on the tabletop is transmitted to the wall screen and executed there. The device connection and file transmission are shown in Fig.4.

The interface can be used in the same way in other devices, except selecting operations change accordingly. Because we cannot select a keyboard and mouse or files on laptop or mobile device without a touch panel, we suggested an additional interface by adding Select functions to the Windows shell menu. With the right click of a mouse, the user can find a file select menu or a keyboard/mouse connection menu. Fig.5 shows Select operations in a laptop (without a touch panel) and mobile device (with a touch panel).

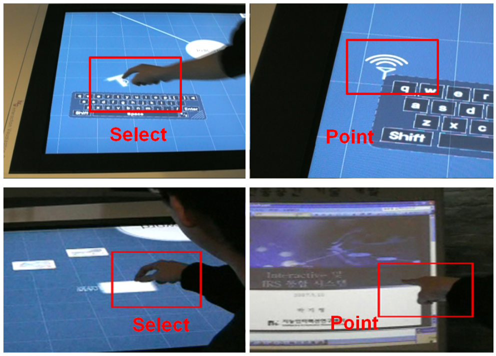

Fig. 4. Device connection(upper) and file transmission(lower)

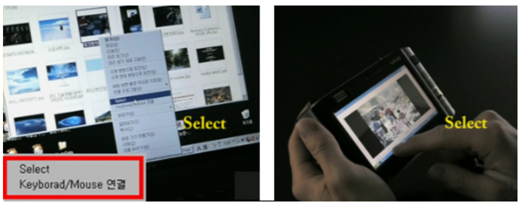

Fig. 5. Select operations in the laptop and touch sensitive mobile device (left: laptop without touch panel; right: UMPC with touch panel)

After selecting the input device or data either by click or touch, the user can connect his/her own device with the indicated device with the same pointing gesture.

5 User Study

We performed a preliminary qualitative user study in the above-mentioned environment to verify our expectation that the suggested interface would improve users' interaction in a multiple device environment.

A total of twelve subjects (8 males, 4 females) recruited within the Korea Institute of Science and Technology participated in the experiment. The age distribution was from early twenties to late thirties. All of them have used and were familiar with popular computing devices. In contrast, most participants had little or no exposure to the tabletop and interactive wall displays.

We conducted three sets of experiments for each participant. Before the main experiments, we asked each participant to "send one picture file on the tabletop to one of wall screens" without any information about the suggested interface. We then explained the concept of the suggested interface to all participants for a few minutes and then carried out the main experiments. Three experiments, each of which consists of several tasks, were designed so that the participants would be exposed to different applications. Of course, each participant was trained for about 15 minutes on how to use the devices in the experimental environment – the tabletop system, Windows shell menu of laptop, and media player of UMPC.

In Experiment 1, the participant was asked first to create a virtual keyboard and mouse and to connect them to one of wall screens. The participant was then asked to transmit a presentation file on the tabletop to the connected wall screen, and modify some part of the file using the connected virtual keyboard and mouse. Finally, the participant was asked to control the presentation slide with bare-hand gestures. Experiment 2 is similar to Experiment 1 except it used the laptop instead of the tabletop to connect to and control the wall screen. The participant was asked to send the word file on his/her laptop to the wall screen and modify it with the laptop. In Experiment 3, the participant was first asked to play the video clip on his/her UMPC. The participant was then asked to pause the video clip on his/her UMPC, and resume it on one of wall screens. For this experiment, we specially developed a media player that enables resuming the multi-media file transmitted from the other device at the position the user has paused.

After these experiments, we asked each participant the following simple questions: 1) how easy was it to understand and use the suggested interface?; 2) how would you prefer the suggested interface to the existing interface and method?; and 3) what are the problems of the suggested interface if any exist?

In the pre-experiment, all of participants only selected the file and could not proceed any further. However, with a hint about using hand gestures, three participants exactly performed selecting and pointing gestures. In the main experiments all of the participants easily understood the concept of the interface, and all but one participant completed several tasks in the three experiments successfully without failure in connecting devices and controlling the data. Ten of the twelve participants preferred the proposed method to the traditional ones of using physical cables, e-mail/FTP, and mass storage devices. Two participants gave comments about the suggested interface: "It could also be a cumbersome process to setup several programs in each device to use the interface," and "Doing hand gestures in a public space makes me a little bit uncomfortable." Experimental results indicate that first-time users can easily use the suggested interface with brief explanations about the concept.

6 Conclusion and Future Works

The suggested gesture-based interface enables users to manage connections and handle information data intuitively and naturally in a multiple device environment. We fully implemented the suggested interface in an intelligent meeting room consisting of a tabletop, interactive wall displays, and other popular devices such as laptops and mobile devices. From the results of the user study, we knew that the concept of the suggested interfaces in different types of devices is easily accepted by users. Although there are a few comments concerning the use of the interface, we hope that this interface and the possible following works can significantly improve users' interaction with multiple devices in tabletop and interactive surface environments.

We are now working on combining this interface with various types of command gestures to easily handle information data and control devices. As one experiment participant suggested, we are also considering extending the suggested interface to control every day machines, such as home electronics and appliances. Finally, we plan to conduct extensive user studies to better understand the issues and problems in using the gesture-based interface in a ubiquitous environment.

Acknowledgments. This research was funded by Korea Institute of Science and Technology.

References

1. Ayatsuka, Y., Matsushita, N., Rekimoto, J.: Gaze-Link: A New Metaphor of Real-World Oriented User Interface. IPSJ Journal 42(6), 1330–1337 (2001)
2. Ayatsuka, Y., Rekimoto, J.: tranSticks: physically manipulatable virtual connections. In: Proc. CHI 2005, pp. 251–260. ACM Press, New York (2005)
3. Borchers, J., Ringer, M., Tyler, J., Fox, A.: Stanford interactive workspaces: a framework for physical and graphical user interface prototyping. IEEE Wireless Communications 9, 64–69 (2002)
4. Citrix's ICA, `http://www.citrix.com/technology/icatech.htm`
5. Everitt, K., Shen, C., Ryall, K., Forlines, C.: MultiSpace: Enabling Electronic Document Micro-mobility in Table-Centric, Multi-Device Environments. In: Proc. TableTop 2006, pp. 27–34 (2006)
6. Iwasaki, Y., Kawaguchi, N., Inagaki, Y.: Touch-and-Connect: A Connection Request Framework for Ad-Hoc Networks and the Pervasive Computing Environment. In: Proc. IEEE International Conference on Pervasive Computing and Communications, pp. 20–29 (2003)
7. Johanson, B., Hutchins, G., Winograd, T., Stone, M.: PointRight: Experience with Flexible Input Redirection in Interactive Workspaces. In: Proc. UIS 2002, pp. 227–234 (2002)
8. Kova, J., Peer, P., Solina, F.: Human skin color clustering for face detection. In: Proc. IEEE EUROCON 2003, vol. 2, pp. 144–148 (2003)
9. Lee, H., Jeong, H., Lee, J., Yeom, K., Shin, H., Park, J.: Select-and-Point: A Novel Interface for Multi-Device Connection and Control based on Simple Hand Gestures. In: Extended Abstract. CHI 2008, pp. 3357–3362. ACM Press, New York (2008)
10. Letessier, J., Bérard, F.: Visual tracking of bare fingers for interactive surfaces. In: Proc. UIS 2004, pp. 119–122 (2004)
11. Lucas, B., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proc. Imaging understanding workshop, pp. 121–130 (1981)
12. Microsoft Corporation, Remote Desktop Protocol (RDP), Features and Performance White Paper (2000)
13. Miller, R., Myers, B.: Synchronizing clipboards of multiple computers. In: Proc. UIS 1999, pp. 65–66 (1999)
14. Park, J., Yeom, K., Lee, J.: Intelligent and Responsive Spaces: Integrated Design of Ambient Physical Spaces and Information Spaces. Korean Journal of Information Society 26(3), 15–25 (2008)
15. Park, J., Yeom, K., Ha, S., Park, M., Kim, L.: An Overview of Intelligent Responsive Space Technology in Tangible Space Initiative. In: Fishkin, K.P., Schiele, B., Nixon, P., Quigley, A. (eds.) PERVASIVE 2006. LNCS, vol. 3968, pp. 223–232. Springer, Heidelberg (2006)
16. Rabiner, L.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proc. IEEE 77(2), 257–286 (1989)
17. Rekimoto, J.: Pick-and-Drop: A Direct Manipulation Technique for Multiple Computer Environment. In: Proc. UIST 1997, pp. 31–39 (1997)
18. Rekimoto, J., Ayatsuka, Y., Kohno, M.: SyncTap: An Interaction Technique for Mobile Networking. In: Proc. Mobile HCI 2003, pp. 104–115 (2003)

19. Rekimoto, J., Ayatsuka, Y., Kohno, M., Oba, H.: Proximal interactions: A direct manipulation technique for wireless networking. In: Proc. INTERACT 2003, IFIP, pp. 511–518 (2003)
20. Richardson, T., Stafford-Fraser, Q., Wood, K., Hopper, A.: Virtual Network Computing. Journal IEEE Internet Computing 2(1), 33–38 (1998)
21. Streitz, N., Tandler, P., Muller-Tomfelde, C., Konomi, S.: i-LAND: An Interactive Landscape for creativity and Innovation. In: Proc. CHI 1999, pp. 120–127. ACM Press, New York (1999)
22. Swindells, C., Inkpen, K., Dill, J., Tory, M.: That one there! Pointing to establish device identity. In: Proc. UIS 2002, pp. 151–160 (2002)
23. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proc. CVP 2001, pp. 511–518 (2001)
24. Wigdor, D., Shen, C., Forlines, C., Balakrishnan, R.: Table-centric interactive spaces for real-time collaboration. In: Proc. AVI 2006, pp. 103–107 (2006)
25. X Window System, `http://www.opengroup.org/tech/desktop/x/`
26. Zigelbaum, J., Kumpf, A., Vazquez, A., Ishii, H.: Slurp: tangibility spatiality and an eyedropper. In: Proc. CHI 2008, pp. 2565–2574. ACM Press, New York (2008)