vib-connect : A Device Selecting Interface Using Vibration

Hiroshi Nakahara
Faculty of Environmental and Information Studies
Keio University
5322 Endoh,252-8520, Japan
Email : rossi@ht.sfc.keio.ac.jp

Takahiro Nozawa
Takuro Yonezawa
Jun’ichi Yura
Hideyuki Tokuda
Graduate School of Media and Governance
Keio University
5322 Endoh, 252-8520, Japan
Email : {takahiro,takuro,yura,hxt}@ht.sfc.keio.ac.jp

Abstract—This paper proposes an intuitive device selecting interface for device collaboration. Recent progress in Information Technology allowed various devices to join wireless network, and as a result, various ways of device collaboration and services became possible. However, interfaces for selecting device are still complicated for end-users and far from being intuitive. To solve this problem, we propose “vib-connect”, an interface that enables users to select device intuitively by pasting a vib-connector, a small vibration-based device.

I. INTRODUCTION

Due to the recent progress of information technology, many devices have joined the wireless network today, and collaborating devices has become more familiar to us. As internet of things expanded and became more complicated, various methods for device collaboration have been proposed.

Before the appearance of the wireless networking technology, cables had the role of device selecting interface. “Plugging” motion meant to select the device as the member of device collaboration, and the presence of the cable visualized the relationship of collaborating devices. However, a cable makes distance limit to device collaboration, and plugging cable to devices all over the room is obviously nonsense. After the wireless networking technique was brought to device collaboration, users were released from the troubles of obstacles and distance limit. Interfaces such as listing up connective devices on the screen and making the users to select became common, but these interfaces are difficult to tell if the users are selecting the correct device. Also, after collaborating devices, recognizing the relationship of collaborating device is difficult. Since types and the numbers of collaborating devices are increasing and becoming more complicated for end users, correspondence between the virtual world and the real world must be visualized clearly.

II. REQUIREMENTS FOR INTUITIVE AND CAPABLE DEVICE SELECTING INTERFACE

Problems and requirements for intuitive, capable device selecting interface can be described as follows :

1) Visualizing correspondence between virtual world and the physical world : Interfaces such as listing up connective devices on the screen and making users to select became common, but these interfaces are difficult to tell if the users are selecting the correct device. Also, after collaborating devices, recognizing the relationship of collaborating device is difficult. Since types and the numbers of collaborating devices are increasing and becoming more complicated for end users, correspondence between the virtual world and the real world must be visualized clearly.

2) Being free from obstacles and distance limit : Even if you used cables in device collaboration, to visualize correspondence between the virtual world and the real world, the cable makes a distance limit “as far as the cable reaches” to device collaboration. Since connective devices are spread all over the room today, the device selecting interface must be free from the troubles of obstacles and distance limits.

3) Not occupying the I/O ports : In order to deal with the previous two problems, there is a way to use I/O ports (ex. USB ports, memory sticks). For example, in wivia[1], users insert a small device to the USB port, and the screen will be shared. However, this had a problem that the limited I/O ports (In this case, USB ports) were occupied. Since most of the user devices do not have rich resources, device selecting interface must not occupy the limited I/O ports.

Some studies in the past focused on these problems, but they do not cover all of the requirements. For example, in touch-and-connect[2], users select device by pushing the button which is put nearby the devices. This interface satisfies requirement 2) and 3), but after the devices are being collaborated, users are not able to recognize the relationship of collaborating devices. In u-Photo[3], visual markers are pasted on the devices, and users were able to select devices to collaborate by selecting them over camera. This interface satisfies requirement 1) and 3), but this interface does not satisfy requirement 2).
To satisfy these requirements, we propose vib-connect. In vib-connect, users select device intuitively by pasting a small device called “vib-connector”. By the presence and the color of vib-connector, correspondence between the virtual world and the real world is visualized clearly, and the users can easily recognize the relationship of collaborating devices. Also, as vib-connector communicates over wireless network and can be pasted anywhere on the user device, it does not occupy the I/O ports of the user device, therefore, the users are free from troubles such as distance limit in collaborating devices.

In section 3, we will explain the details of the concept and the design of vib-connector.

III. VIB-CONNECT

A. Concept

We assume two types of devices, service device (I.E. Display, speaker, printer etc.), and user devices (I.E. Laptop computers, PDA, smart phones etc.). In vib-connect, users use a small device called “vib-connector”, to select these two devices to collaborate. As illustrated in Fig.1, users only need to: (a) touch the service device with vib-connector, and (b) paste the vib-connector to the user device. vib-connector specifies the user device, which the vib-connector is pasted, by causing vibration to the user device. Then, the devices selected by the vib-connector starts to collaborate. In order to disconnect the device collaboration, users simply take off vib-connector from the user device.

B. Design and Implementation

vib-connector consists of RF-ID reader, vibration motor, color-LED, and switch (to detect itself being pasted). In vib-connect, there are 2 steps to start the device collaboration:

1) TOUCH - Relating service and vib-connector: In the default state, the color of vib-connector is white. Each service device has a RF-ID tag pasted on, and each of them has an unique color. When users touch the RF-ID tag with vib-connector, vib-connector reads color information from the RF-ID tag, and vib-connector glows in the same color with the RF-ID tag. Also, specific vibration pattern is read from the RF-ID tag and saved to vib-connector.

2) PASTE - Specifying the user device: Detecting itself being pasted, vib-connector starts to vibrate in the vibration pattern, which is saved in the vib-connector itself. User devices tries to recognize the vibration pattern by the capable resource, a built-in microphone or acceleration sensor. Recognizing the vibration pattern, the user device interrogates the information server about information of the service device. Receiving the

Fig. 1. Selecting device in vib-connect

information of service device, the user device opens a connection to the service device.

Fig.2 is the system architecture of vib-connector. Vibration pattern and color information of the tag is saved in the RF-ID tag, which is pasted on the service device. As soon as these information are read and saved in vib-connector, LED glows in the same color with the RF-ID tag, according to the color information which was read from RF-ID tag. When the switch detects vib-connector itself being pasted, vib-connector starts to vibrate in the vibration pattern, which is saved in the vib-connector itself. Recognized vibration pattern will be interrogated to the information server, by the user device. Receiving the information of service device as a reply from the information server, the user device tries to connect to the service device. To close the connection, user simply have to remove vib-connector from the user device.

IV. CONCLUSION

In this paper, we proposed an intuitive device selecting interface, “vib-connect”. In vib-connect, users paste a small vibration-based device called “vib-connector”, to select devices to collaborate. vib-connector characterizes the microphone and acceleration sensor of the user device and specifies devices by vibration, and enables devices to connect. This feature gives an advantage to users in which single ports do not have to be occupied. Also, by the presence and the color of vib-connector, correspondence between the virtual world and real world is visualized, and the users are able to recognize the relationship of the collaborating devices.

ACKNOWLEDGMENT

This work was supported by NICT as a part of the Dynamic Network Project.

REFERENCES