Spinner:
A Simple Reconfigurable Haptic Interface

Shigeru Kobayashi and Masayuki Akamatsu
International Academy of Media Arts and Sciences
3-95, Ryoke-cho
Ogaki City, Gifu 503-0014 Japan
+81-584-75-6600
{mayfair, aka}@iamas.ac.jp

ABSTRACT
This paper reports on our recent development of a reconfigurable haptic user interface. We created a system that consists of a dial type of controller 'Spinner,' and the virtual contoller objects. One physical controller corresponds to one virtual controller on a PC's display device, and a user can freely change the connection on the fly (i.e. associate the physical controller to another virtual controller). 'Spinner' is versatile and can be used for many purposes that need connections between virtual controllers and physical controllers. By using this system, a user can build a reconfigurable haptic user interface.

ACM Classification H5.2 [Information interfaces and presentation]: User Interfaces. - Haptic I/O.

General Terms Design, Human Factors, Performance

KEYWORDS: Reconfigurable, haptic, physical, graphical user interfaces

INTRODUCTION
How to access virtual controllers on a computer screen by limited types of physical controllers (e.g. mice, ASCII keyboards and so on) has been a long-term issue. For example, nonlinear video editing software has many control parameters, and the parameters are represented as virtual controllers (e.g. virtual knobs, sliders, switches and so on).

Operating a virtual controller by a mouse is a popular way, but there are drawbacks. For example, a user can control only one parameter at once, and a user can not feel proper haptic feedbacks. To solve this issue, physical control surfaces for video editing or musical environments have been developed[2][3]. A typical physical control surface has knobs, sliders and buttons. But there is another issue: Typically, the number of virtual parameters on a computer screen is much more than the number of physical controllers. To solve this issue, many physical control surfaces equip assignable physical controllers on their surface, and users can configure their settings of connections from a physical controller to a parameter (or multiple parameters).

However, there is a drawback with this kind of approach: Since the layout of physical controllers is fixed, a user still needs to do a conversion from the position of the physical controller in a physical space to the position of a virtual controller in a virtual space. This conversion might improve with practice, but it still requires time before it can be done automatically.

One possible solution to this issue is making a user interface reconfigurable. For instance, Buxton et al. has proposed reconfigurable interfaces consisting of a touch-sensitive tablet and templates[1]. And recently, 'Pin&Play&Perform' by Villar et al. is a reconfigurable interface as well[8]. We have also tried an approach to create a reconfigurable user interface by a simple method[4]. Since the original 'Spinner,' we have updated it to improve the usability (especially for the identification part, making it a few times faster than the original).

BASIC CONSTRUCTION PRINCIPLES
Figure 1 shows a basic usage of 'Spinner'. Two physical controllers (i.e. dials) are connected to the interface device, and then the interface device is connected to the PC via USB. A user can access a parameter on the PC's screen by rotating the binded dial.

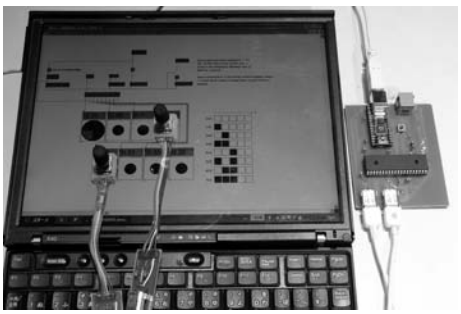

Figure 1: Spinner in operation.

If the person wants to change the binding, they just lift the dial off the display device, then put it on another virtual dial and push the knob to start an identification process. In the identification process, a special pattern for identification is displayed on the display device, and the dial device detects

where it is located. Since a binding is determined on the fly, a user does not have to remember bindings.

IMPLEMENTATION AND RESULTS

Composition

Figure 2 shows the composition of the physical controller and the interface board. A photo IC is mounted on the bottom face of a dial to detect luminance on a display where the dial is located. All information from rotary encoders and sensors are handled by the microcontroller on the interface board.

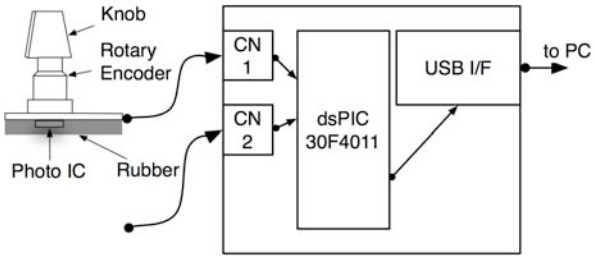

Figure 2: Composition of the physical controller (left) and the I/F board (right).

Figure 3 show examples of graphical user interfaces. Since the PC side software is implemented on a highly customizable Max/MSP platform from Cycling '74, a user can easily modify an existing arrangement to make a new variation or create a new one from scratch[5].

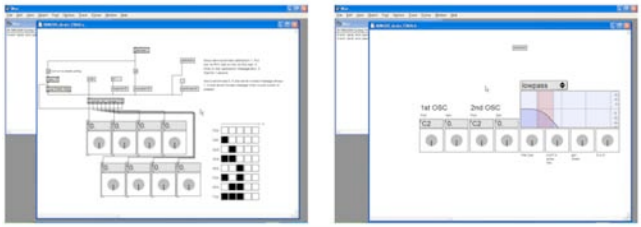

Figure 3: GUI examples.

Identification

The method of identification is very simple. Figure 4 shows the timing chart of an identification process. In each period, a message which shows a current bit index is sent (higher-order bits first), then a pair of identification patterns are presented in sequence at each virtual controller's rectangular region.

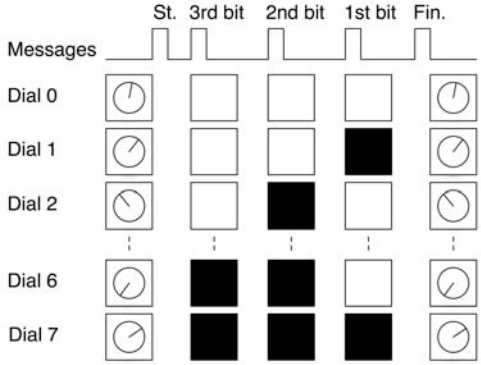

Figure 4: An example of identification pattern.

The identification patterns consist of black (means '1') or white (means '0'). For example, a physical dial detects a pattern like 'black-black-white' (i.e. '1-1-0'), which means that the physical dial is located on the sixth virtual dial, so it will be associated with the sixth virtual dial.

Results

We found that the system is very easy and fun to use, and contains many possibilities. For identification time, about 75ms is required per one period. For example, if a user instantiates 8 virtual controller objects, about 220ms is required for the whole identification process. This is similar to the typical reaction time[6].

CONCLUSIONS AND FUTURE WORK

We have developed 'Spinner' to a state whereby it can be used satisfactorily, and we feel that it holds many possibilities. As a first step, we validated the usefulness with a musical application, but it can be adapted to many applications that need interactions between virtual and physical controllers (e.g. 3D CAD/CAM applications, nonlinear video editing applications and so on). Additionally, since 'Spinner' features a simple identification method which does not require special display devices or touch panels, it's capable of being used with many display devices, such as an LCD (Liquid Crystal Display) or a CRT (Cathode Ray Tube) and so on. In the next stage, we would like to improve 'Spinner' as follows:

- More application platform support (e.g. Java, Processing[7] and so on)
- Wireless connection support.
- Faster and more accurate identification.
- Other types of controllers (e.g. switch, slider).

REFERENCES

1. Buxton, W., Hill, R. & Rowley, P. Issues and techniques in touch-sensitive tablet input. Computer Graphics, 19(3), In *Proceedings of SIGGRAPH'85*, 1985, pp. 215–223.

2. Edirol Corporation. http://www.edirol.com/products/midi.html, 2005.

3. JLCooper Electronics. http://www.jlcooper.com/pages/controllers.html, 2005.

4. Kobayashi, S., Akamatsu, M. Spinner: A Simple Approach to Reconfigurable User Interfaces. In *Proceedings of NIME05*, 2005, pp. 208–211.

5. O'Sullivan, D., Igoe T. *Physical Computing.* Muska & LipmanPremier-Trade, 2004

6. Raskin, J. *The Humane Interface.* Addison-Wesley, 2000.

7. Reas, C., Fry, B. Processing: a learning environment for creating interactive Web graphics. In *Proceedings of the SIGGRAPH 2003*, 2003.

8. Villar, N., Lindsay, A., Gellersen, H. Pin&Play&Perform: A rearrangeable tangible interface for musical composition and performance. In *Proceedings of NIME05*, 2005, pp. 188–191.