# Surfboard: Keyboard with Microphone as a Low-cost Interactive Surface


*Jun Kato[1,2], Daisuke Sakamoto[1], Takeo Igarashi[1,2]*

[1]JST ERATO IGARASHI Design UI Project
1-28-1, Koishikawa, Bunkyo-ku, Tokyo, 112-0002, Japan
{kato | sakamoto}@designinterface.jp

[2]Graduate School of Information Science and Technology, The University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-0033, Japan
takeo@acm.org



## ABSTRACT

We introduce a technique to detect simple gestures of "surfing" (moving a hand horizontally) on a standard keyboard by analyzing recorded sounds in real-time with a microphone attached close to the keyboard. This technique allows the user to maintain a focus on the screen while surfing on the keyboard. Since this technique uses a standard keyboard without any modification, the user can take full advantage of the input functionality and tactile quality of his favorite keyboard supplemented with our interface.



**ACM Classification:** H5.2 [Information interfaces and presentation]: User Interfaces - Input devices and strategies.

**General terms:** Design**,** Human Factors

**Keywords:** Keyboard, microphone, low-cost, interactive surface


## INTRODUCTION

As a set of physical buttons laid out flat, a keyboard can be thought of as a bumpy surface. We propose a technique named *Surfboard* which aims to augment the input capability of a keyboard by recording and analyzing sounds produced when the user lightly touches the keyboard and moves his fingers horizontally over it. We named this action "surfing". The keyboard is a standard input device and we are not the first to augment it. Dietz et al. presented a practical pressure-sensitive keyboard [1] which provides pressure information supplemented with pressed keys. Block et al. presented a touch-display keyboard [2], each of whose key tops is augmented with a touch sensor and a graphics display on its surface. There is also a commercial product that covers a keyboard with a multi-touch surface [3]. Unlike these previous attempts, *Surfboard* augments a keyboard with a monaural microphone, making its hardware setup simple and inexpensive. It adds an additional operation modality, "surfing" to standard keyboards without changing their physical properties. Simple gestures of *Surfboard* allow the user to maintain a focus on the screen during a surfing gesture. Since surfing and typing happen in same place, the user can seamlessly continue touch typing after surfing.



*UIST'10,* October 3–6, 2010, New York, New York, USA.
ACM 978-1-4503-0271-5/10/10.



Figure 1. "Surfing" gesture on a keyboard

Compared with a standard keyboard operation, surfing does not take time to find an intended key among a lot of keys. A mouse and a touchpad are generally used for the cursor operation and cannot be used with fingers put in the home position of the keyboard. These comparisons suggest that *Surfboard* is good for tasks which are often required during text entry or tasks which must be done in short time when the need arises.

## SURFING ON A KEYBOARD

*Surfboard* can recognize whether the user is surfing on a keyboard or not (Figure 1). When the user surfs on a keyboard, it can distinguish left to right or right to left surfing directions. We have not yet tested other motions like drawing circle or more complex shapes because they would be difficult to execute on a keyboard which is generally horizontally long.

*Surfboard* monitors the user's typing activity and its gesture recognizer automatically stops when the user is typing something. As a result the user need not explicitly switch between surfing and typing. Additionally, surfing can be used in combination with pressing modifier keys like Shift, Control and Alt to simulate multi-touch gestures. For example, an application may zoom in when a Shift key is pressed and surfing is detected from left to right, and zoom out when surfing from right to left.

## PROTOTYPE IMPLEMENTATION

Our prototype implementation uses a monaural microphone attached close to the keyboard of a personal computer. It


Figure 2. Microphones attached close to keyboards

needs to be attached near the right or left edge of the keyboard to distinguish surfing direction effectively. Fortunately at present, many laptop computers are equipped with a microphone, and we confirmed that *Surfboard* works at a quiet office and a noisy cafeteria with built-in microphones that are located at several different positions (Figure 2).

The users' surfing sounds on the keyboard are captured at a sampling rate of 44.1 kHz and processed with a Fast Fourier Transform function to get amplitude information by frequencies. Static ambient noise is recorded beforehand and is subtracted from the raw data. The process of surfing recognition consists of two phases of Naïve Bayes classification (Figure 3). The first phase detects whether the user is surfing or not in real-time with resampled low resolution sound data. A set of sound data is divided into 50 frequency bands with amplitude information varying from zero to five. The second phase starts when the first classifier recognizes the current sound as surfing. Sound is recorded until the end of surfing is detected. At the end, all recorded data is resampled along the time axis to have normalized length which represents change in sound amplitude during surfing. The recorded data is resampled into the former and the latter part, each of which holds amplitude information of five frequency bands varying from zero to five.

The user needs to train the classifiers in two steps before starting to use *Surfboard*. First, the user is asked to press the Shift key and surf on a keyboard repeatedly for twenty seconds. Second, the user is asked to surf from left to right and right to left for ten times each. After this one minute training process, the system robustly detects the user's surfing gestures. During the general use of *Surfboard*, if the classifiers mistakenly recognize a gesture, the user can correct the last result to improve detection accuracy.

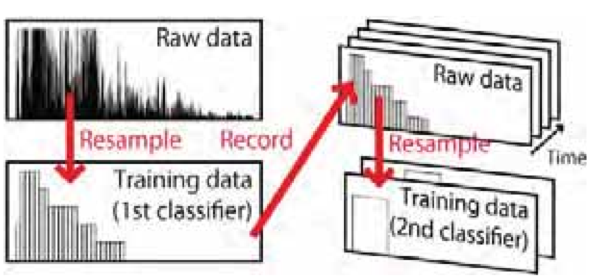


Figure 3. Process of gesture recognition

## LIMITATION AND FUTURE WORK

We have not taken a formal user test, so we recognize need for it to assess robustness of our interface quantitatively against different noise levels.

In our future work, we will explore the use of multiple microphones, which we expect to provide more robust recognition results. Specifically, the use of a stereo microphone should be investigated first since it can be easily carried out on many laptop computers which have capability to capture stereo sound. One of the microphones could be located distant from the keyboard (e.g., on top of the display) to record ambient sound for dynamic noise cancelling. With a microphone array, it might be even possible to estimate the rough position of the hand similar to PingPong Plus [5] which detects the position of a ball hitting a game table.

We will also investigate use of duration information of a surfing action. It could be used to represent the strength of command like a scaling factor in the case of zooming. This is similar to Scratch Input [4], which uses scratching sounds on a flat surface, in the sense that acoustically unique gestures are recognized.

This research could also benefit from improvements to its supervised learning process. The user interface for the training process needs improving. The choice of machine learning technique directly affects the robustness of the gesture recognition. Blowable user interfaces [6], which localizes where the user is blowing on the screen with a monaural microphone, uses k-Nearest Neighbor classification. Comparing several techniques may achieve better performance.

## REFERENCES

1. Dietz, P. H., Eidelson, B., Westhues, J., and Bathiche, S. A practical pressure sensitive computer keyboard. In *Proc. of UIST'09*, ACM, NY, 2009, pp.55-58.
2. Block, F., Gellersen, H., and Villar, N. Touch-display keyboards: transforming keyboards into interactive surfaces. In *Proc. of CHI'10*, ACM, NY, 2010, pp.1145-1154.
3. FingerWorks: http://fingerworks.com/
4. Harrison, C. and Hudson, S. E. 2008. Scratch input: creating large, inexpensive, unpowered and mobile finger input surfaces. In *Proc. of UIST'08*, ACM, NY, 2008, pp.205-208.
5. Ishii, H., Wisneski, C., Orbanes, J., Chun, B., and Paradiso, J. 1999. PingPongPlus: design of an athletic-tangible interface for computer-supported cooperative play. In *Proc. of CHI'99*, ACM, NY, 1999, pp.394-401.
6. Patel, S. N. and Abowd, G. D. 2007. Blui: low-cost localized blowable user interfaces. In *Proc. of UIST'07*, ACM, NY, 2007, pp.217-220.