

# Multi-Touch Table as Conventional Input Device

Andreas Dippon<sup>1</sup>, Florian Echter<sup>2</sup>, Gudrun Klinker<sup>1</sup>

<sup>1</sup> Technische Universität München, Dept. of Computer Science  
Boltzmannstr. 3, 85748 Garching, Germany  
{dippona, klinker}@in.tum.de

<sup>2</sup> Munich University of Applied Sciences, Dept. of Computer Science  
Lothstr. 34, 80335 München, Germany  
[florian.echter@hm.edu](mailto:florian.echter@hm.edu)

**Abstract.** In order to improve the functionality of multi-touch devices, the possibility of using them as input devices for other computers needs to be reviewed. The idea is, to get rid of many different peripherals (e.g. keyboard, mouse, multi-touch pad) by using a single multi-touch display. Furthermore the display can be used as an additional monitor to show for example toolbars, which can be directly manipulated through multi-touch gestures.

We implemented a prototype, which provides an adaptive keyboard that shows shortcut keys for different applications, a multi-touch pad as well as the option to drag&drop standard Windows widgets onto the multi-touch table, which can be controlled by direct touch input. A user study was conducted to test the current system and to get information about the further approach to this concept.

Keywords: multi-touch, input device, libTISCH, virtual keyboard

## 1 Introduction

The idea of our project is, to bring big multi-touch screens into daily office life in the future. Instead of a normal desk, a multi-touch screen is used. The screen is used as an input device for a normal desktop computer and replaces all conventional input devices. A keyboard can be shown, as well as a multi-touch pad, a drawing table, a piano, a mixing desk and so on. At the same time it can be used as an additional monitor, so that programs and windows can be dragged onto the screen and be controlled with direct touch input. Several advantages are depicted in the following list:

- changeability: you can simply change the shown virtual input devices, without having to rearrange your workspace. Therefore you can switch easily between different tasks, always using the appropriate input device.

## 2     **Andreas Dippon, Florian Echtler, Gudrun Klinker**

- adaptivity: the virtual input devices can adapt to the current language, program, etc. For example, the virtual keyboard can show the correct keyboard layout for different languages, or provide the user with additional information (e.g. when pressing the ctrl key in a program, the shortcuts of functions are shown on the keys).
- scalability: the virtual input devices can be rescaled to meet each user's demands.

Yet there are currently several drawbacks:

- haptic feedback: currently, multi-touch screens don't provide any haptic feedback, which makes it harder to write on an onscreen keyboard without looking at it, than on a physical keyboard.
- size and costs: big, flat multi-touch screens are still very expensive and therefore not applicable as standard office desks.

After reviewing related work, we describe our prototype implementation and then show the results of the conducted user study.

## **2     Related Work**

The inverse approach of augmenting keyboards with touch sensitivity and small displays in each key was already presented by several companies and researcher. A first commercial product with small displays in each key was presented with the Optimus Maximus Keyboard by Art Lebedev Studio[1]. Here, the keyboard layout can be changed via software and can show different languages or program related icons. The *Touch-Display Keyboard* project by Block et al.[2] improves this concept by adding touch sensors to each key and using the keyboard as an additional display for the computer. A quite similar project is the *Adaptive Keyboard* from Microsoft, which was used in the Student Innovation Contest at UIST 2010[3]. The touch sensitivity of the keys is removed in this keyboard, but an additional small multi-touch area is added on top of the keyboard. The keys are transparent and the screen which is used for the multi-touch area continues under the keys, so it can be used to display different layouts on the keyboard.

Another related project, which deals with the idea of combining a virtual desktop with a real one, is the Curve Project by Wimmer et al.[4]. They constructed a prototype for an interactive desktop, based on the concept of the DigitalDesk by Wellner[5]. The *Curve* blends a horizontal and a vertical interactive surface, which takes existing ergonomics research into account.

## **3     Prototype Implementation**

In order to test the introduced concept, a prototype was implemented on a multi-touch screen. For the multi-touch support we used the libTISCH library by Echtler[7]. The prototype features an adaptive software keyboard, a multi-touch pad and an area which can be used for direct touch input.

### Direct Touch

The concept of direct touch input is, that users can drag windows or toolbars onto the multi-touch screen and use their fingers instead of a mouse or a touchpad. For example, the user can navigate through the file system or change colors in a graphics editor by directly touching the according buttons.

### MTPad

The implementation of the multi-touch pad (or MTPad) in our prototype is capable of moving the mouse cursor, left and right clicking and scrolling by using multiple fingers. For convenience, the MTPad looks like a standard touchpad of a notebook. The MTPad also takes advantage of the large area of the multi-touch display. Other than on a standard touchpad, the user doesn't have to move the cursor in several small steps, because he can start on the touchpad and just keep moving his finger on the multi-touch display till the cursor has reached the desired position. This effect doesn't interfere with the direct touch functionality, because each tracked touch is always related only to the area where it was sensed first.

### Keyboard

The most important aspect of our concept is the software keyboard. To keep the adaption level of the user low, we decided to use the shape of a conventional keyboard, instead of testing new designs. The implemented keyboard supports different key layouts and program specific shortcut icons can be shown for several programs: *Inkscape*<sup>1</sup>, *Windows Calculator* and *Windows Notepad* (see figure 1).



Fig 1. The adaptive keyboard with the shortcuts of Inkscape while pressing CTRL.

## 4 User Evaluation

In order to get a first impression of how people react to the introduced concept, a small user study was conducted to get some information about the acceptance of the system. Our first approach was to do a study of the entire concept. Although the outcome of such a test would be very valuable, there are several problems conducting the test. In order to test the concept for everyday work, the hardware would have to be

<sup>1</sup> <http://inkscape.org/>

integrated into a normal desk, such that the test person could sit and work there normally. Furthermore, like in the test conducted by Wigdor et al. on *Living with a tabletop*[8], the test subject would have to use the system for a long time to get used to it and in order to get significant test results.

Instead we decided to conduct a shorter study about the implemented keyboard functions. The goal was to get information about the utility of the adaptivity of the keyboard. The setup of the study consisted of a self-built 40" FTIR multi-touch table as the horizontal display and a vertical 42" TV as the main monitor.

In the study, the users had to do two exercises in *Inkscape*, which were designed to first get in touch with the system and then focusing on the utility of overlaid shortcuts on the keyboard. In the first task, the users had to change the color of different objects by using direct touch on the color toolbar of the program, which could be dragged onto the multi-touch screen. In the second task, the users had to solve simple riddles by using the shortcuts of a set of commands on a set of objects, to get a certain object. On the left side of the TV screen the set of objects the users had to combine was shown, in the middle the set of commands they should use and on the right the object they should create. All shortcuts of *Inkscape* were visualized on the corresponding keys on the keyboard during the tests. After the tasks, the users participated in an interview about their impressions of the system. Eight users participated in the study, mostly male computer science students with an average age of 28. More than half of the users thought that the adaptive keyboard is very useful. Especially the shortcut icons were considered very useful in order to get used to non frequently used shortcuts. Two users also mentioned that they could imagine separate keyfields containing the shortcuts of a program. Especially the participants, who are frequently using programs with additional toolsets, like graphics editors, game editors or electronic layout editors, could imagine, that the use of direct touch input would improve their efficiency at working with those programs. Nearly all users stated that they could imagine using the system without haptic feedback for work where they don't have to type longer texts, but for writing or programming they would prefer a keyboard with haptic feedback. More than half of the participants also mentioned that another important aspect that needs to be improved is the sensitivity and calibration of the system, so that unintended touches or mouse clicks are not triggered.

## 5 Future Work

**Adaptive Keyboard:** The shortcut icon concept could support more programs and additional language layouts could be implemented. Another aspect is to make the keyboard more flexible, so that each user can define the shape individually (e.g. for users who prefer natural keyboards). The next evaluation step for the shortcut icons could be the comparison between different layouts of the shortcuts (e.g. ordered by menu appearance, groups of similar functions, etc.) and also the comparison between shortcuts on the keyboard versus shortcuts on additional buttons.

**Multi-touch input:** A big challenge in the future will be the support of multi-touch gestures in standard programs. The task here lies in finding fitting gestures to control programs by multi-touch gestures rather than just mapping the gestures to conventional commands.

**Concept:** The whole concept of a multi-touch desk in addition to the normal computer screen could be evaluated in a long term study, so that the participants get used to the system in their daily work, where they could eventually identify any drawbacks of the system.

## 6 Conclusion

In our research, the concept of using multi-touch surfaces as input devices for other computers has been introduced. By using a multi-touch surface, which is built into a standard desk, conventional input devices could be replaced through this concept. In order to get a first impression of the usability of the concept, a prototype was implemented. In the prototype, an adaptive keyboard, a multi-touch pad and a direct touch area was realised. The adaptive keyboard featured the visualization of shortcut icons for several programs. The prototype was then evaluated in a small user study to gather information about problems of the concept and also to get suggestions for improving the system.

## 7 References

1. Art Lebedev Studio, <http://www.artlebedev.com/everything/optimus/>
2. Block, F., Gellersen, H., Villar, N.: Touch-Display Keyboards: Transforming Keyboards into Interactive Surfaces. In CHI '10: Proceedings of the 28th International Conference on Human Factors in Computing Systems, pp. 1145–1154, New York, USA. ACM (2006)
3. UIST 2010 Student Innovation Contest, [http://www.acm.org/uist/uist2010/Student\\_Contest.html](http://www.acm.org/uist/uist2010/Student_Contest.html)
4. Wimmer, R., Hennecke, F., Schulz, F., Boring, S., Butz, A., Hußmann, H.: Curve: Revisiting the Digital Desk. In NordiCHI 2010: 6th Nordic Conference on Human-Computer Interaction, New York, USA. ACM (2010)
5. Wellner, P.: The DigitalDesk Calculator: Tangible Manipulation on a Desk Top Display. In UIST '91: Proceedings of the 4th annual ACM Symposium on User Interface Software and Technology, pp. 27–33, New York, USA. ACM (1991)
6. Bi, X., Grossman, T., Matejka, J., Fitzmaurice, G.: Magic Desk: Bringing Multi-Touch Surfaces into Desktop Work. CHI 2011 Conference Proceedings (in Press): ACM Conference on Human Factors in Computing Systems (2011)
7. Echtler, F., Klinker, G.: A Multitouch Software Architecture. In Proceedings of NordiCHI 2008, pp. 463–466 (2008)
8. Wigdor, D., Perm, G., Ryll, K., Esenther, A., Shen, C.: Living with a Tabletop: Analysis and Observations of Long Term Office Use of a Multi-Touch Table. Second Annual IEEE International Workshop on Horizontal Interactive Human-Computer Systems (2007)