Tool Support for Developing Scalable Multiuser Applications on Multi-Touch Screens

Ioannis Leftheriotis
Ionian University
Tsirigoti Sq. 7,
Corfu, Greece
and
Norwegian University of
Science and Technology
Sem Sælands vei 7-9,
Trondheim, Norway
iolef@acm.org

Konstantinos Chorianopoylos
Ionian University
Tsirigoti Sq. 7,
Corfu, Greece
choko@acm.org

Letizia Jaccheri
Norwegian University of Science
and Technology
Sem Sælands vei 7-9,
Trondheim, Norway
letizia@idi.ntnu.no

Abstract

MT (Multi-touch) screens are platforms that enhance multiuser collaboration. In this work, we underline the need for novel interaction techniques and toolkits that allow multi-user collaboration on larger MT surfaces. We present ChordiAction toolkit that makes use of a novel chorded interaction technique allowing simultaneous multi-user interaction on scalable MT applications. We describe the design, the architecture and some efficient customizations practices of the toolkit and show how it can be effectively embedded in an application for multiuser interaction. As a proof of concept, we present some example applications using ChordiAction toolkit showing its potentials and discuss our future plans for further evaluation of this technique.

ITS'12, November 11–14, 2012, Cambridge, Massachusetts, USA.
ACM 978-1-4503-1209-7/12/11.

Author Keywords

Toolkit; chords; multi-touch; large multi-touch screens

ACM Classification Keywords

H.5.2. [Information Interfaces]: User Interfaces-Input devices and strategies; graphical user interfaces; prototyping.

Introduction

MT surfaces have become popular platforms during the last years. In addition, a variety of toolkits that support development for MT devices have been presented [2]. There is no doubt that the MT community is vivid and new toolkits are being developed constantly either by practitioners and hobbyists (e.g., Kivy), or researchers. These multi-touch toolkits usually support multiple operating systems and multiple hardware installations. However, the motivation for this work was the fact that despite the considerable quantity of research directed towards MT technologies, a set of standardized UI components have not been developed yet [4], and more importantly there is a lack of tools dedicated to multiuser interaction on larger screens.

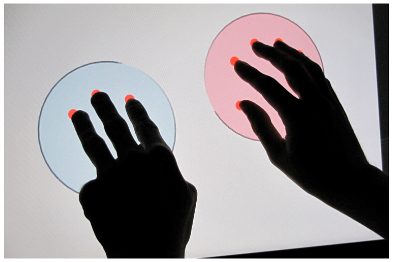


Figure 1: Multiple users can articulate different chords. In this screenshot, two users articulate three fingers chord and five fingers inside the colored reserved areas. Chords are not a novel type of interaction [Grossman], but using chords for multi-user interaction is a novel technique.

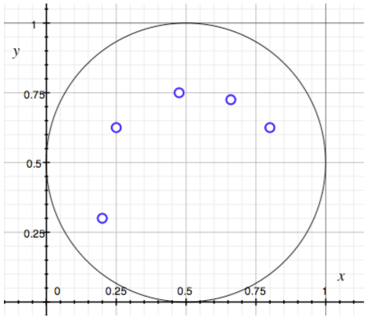


Figure 2: In this graph, the diameter of the circular area is equal to 1. All possible finger-touches and their appropriate positions in relation to the circular chorded area are shown. The positions are based on our own experience. However, positions can be easily changed inside the code.

Although there are Toolkits being focused on MT interaction, only some of them deal with the issue of multi-user synchronous collaboration. In DiamondSpin API, DSFrame DSPanel and DSWindow equivalents of JFrame, JPanel and JWindow of Java are presented. According to Shen et al [5], in order to have multi-user support, multiple menu-bars or concurrent multi-user interaction techniques are needed. In this work, we focus on a novel interaction technique called chords (simultaneous touch of more than one finger on a surface) and we present ChordiAction Toolkit for multi-user interaction on MT surfaces. Hence, the main contribution of our work is: 1) the development of a toolkit that enhances multi-user interaction in larger MT screens, and 2) the proposal and discussion of some characteristics of a multiuser MT toolkit that raised during the development process of our work.

ChordiAction Toolkit

ChordiAction toolkit has been implemented with the PyMT programming framework [1], and it can be easily transferred to any MT Software Development Kit (SDK), because it is based on common MT user interface events, such as touch-down, and touch-up, which are available in any MT toolkit (e.g., Microsoft Surface). Using that event model is a traditional strategy for developing MT applications [1] for a variety of MT toolkits, which make use of this type of events including 2D information such as the coordinates of a touch (e.g. WM_TOUCH).

Design

With ChordiAction toolkit we are dealing two important issues that arise in multi-user MT applications. First, identifying which user has done what on a MT screen and second allowing each user work separately without his actions concerning or even annoying others.

The main idea is the use of a mechanism in order to choose a specific item from a menu without touching the buttons of the menu directly, as well as being able to perform individual interactions within a limited area. A novel multiuser chorded selection technique, where user makes use of a circular chording area that is temporarily (for a number of seconds, e.g. 5 seconds) reserved whenever a user touches the MT screen has been used [3]. In that small area, user has to perform a chord for the selection of the appropriate menu item. With the use of a not-functioning menu bar playing the role of a status indicator (Figure 6), users are able to understand which menu item must be chosen and how many fingers they have to touch on the surface in order to select it.

Multiple users are able to touch different parts of the screen and then different small areas will be temporarily reserved for chorded modifiers accordingly (Figure 1). The reserved area is about the user's hand size, being easy for the user to touch the appropriate number of fingers inside it and thus applying the chord responsible for the selection of the menu item he/she desires. Each time a user makes a selection the appropriate action/function can be activated.

The steps that the system follows for every single user that interacts on the MT screen are the following: a) Initially the systems waits for the first finger touch. b) When this occurs a new chorded area is reserved. c) User has to touch the appropriate number of fingers in the specific chorded area in order to articulate the corresponding chord. d) When user has lifted all his

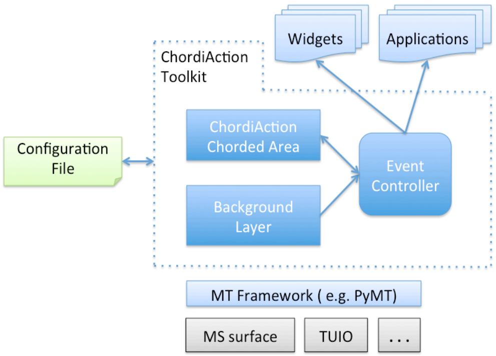


Figure 3: ChordiAction toolkit architecture. The toolkis is developed in PyMT, however, it can be easily migrated to other MT Frameworks.

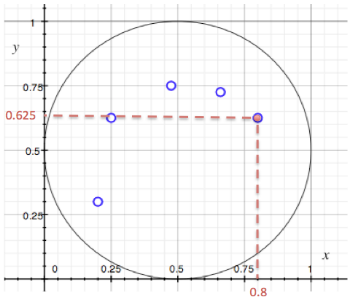


Figure 4: In this example, the circular area pops-up in relation to the fifth finger (pinky). If the pinky enables the chording area, the first touch will be at the (0.8 * circle-diameter, 0.625 * circle-diameter) and the circle will be placed accordingly.

fingers from the screen, the chord interaction area is released. The system then processes the chord and finally, e) The appropriate action or function is enabled.

This five steps strategy of our system stems from the two necessities discussed previously. The system locate the place where the interaction needs to be done, supports us with a method to differentiate what the user needs or which user interact (in our case this method includes chords) and finally, gives the opportunity for undistracted interaction of all users.

Architecture

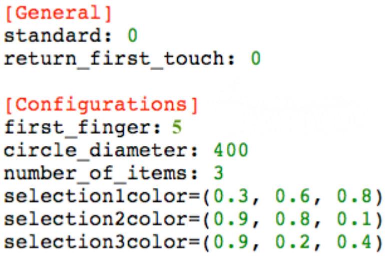
The architecture of the toolkit is based on PyMT event model [1]. Specifically, there are three separate layers and in order to pass from one level to the next, a touch event is needed. In the background layer, user touches the surface in order to appear the chorded interaction area. When this happens an appropriate event is generated, the chorded area pops up and the user is able to articulate the appropriate chord. Finally, a new layer with a new application or widget or action appears (Figure 3).

In larger MT surfaces where there is enough space for multi-user interaction, the system should allow multiple instances of different applications to run concurrently. That is why we propose that even the toolkits themselves should have multiple instances in a multi-user MT environment where multiple applications may run in parallel. In addition, these tools might be able to be applied in different levels of the widget tree of a MT application. However, each ChodriAction toolkit instance must have its own customization file. Additionally, the toolkit can be used for either selecting among different applications or different widgets/functions.

Toolkit Customization

Along with the toolkit, there is a configuration file that can be modified to express developers' preferences and help them integrate the toolkit inside their own applications. For instance, for every chord that has been articulated, developers can have returned either the position of the first finger of each chord or the (0,0) point of the circular area (Figure 5). Additionally, developers are free to choose which finger will be the first to enable a circular chorded interaction area. Whenever user touches the screen, the circular area pops-up and it is placed accordingly to his/her hand. In Figure 4, an example in which the circle is positioned in relation to the pinky finger is depicted.

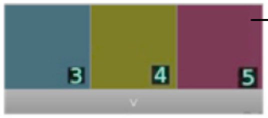
Moreover the toolkit allows different sizes of chord interaction circular areas. Additionally, the number of available selection that can be done using chords can be altered. In the case of four different menu objects for instance, there are four different items and thus four different chords: a) two fingers b) three fingers, c) four fingers and d) five fingers. Note that along with the number of selections, a number of different feedback colors (with which the circular area is painted) must be shown to the user. That is why the toolkit allows developers to chose the appropriate color of the circle for each chord. The colors are in RGB format and can be easily altered. For instance, each time the user makes the third selection (by touching four fingers inside the circular chorded interaction area) the circle turns to red color and thus he/she understands that when all his fingers will be released from the MT surface, the action in red is going to be executed.



```
[General]
standard: 0
return_first_touch: 0

[Configurations]
first_finger: 5
circle_diameter: 400
number_of_items: 3
selection1color=(0.3, 0.6, 0.8)
selection2color=(0.9, 0.8, 0.1)
selection3color=(0.9, 0.2, 0.4)
```

Figure 5: The customization file: the circular area pops up according to the fifth finger, there are 3 possible selections/chords and the circle diameter is 400px.



A not functioning status indicator bar that informs users about the available chords.

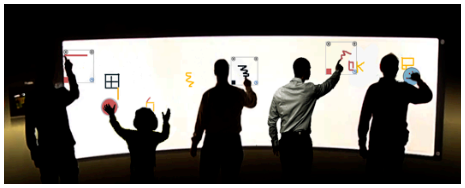


Figure 7: Drawing application that makes use of ChordiAction in combination to personal windows so as to differentiate users' interactions.

Example applications

We are working on the development of collaborative applications in order to evaluate empirically our chorded technique. We have already developed a collaborative dot-to-dot drawing application where multiple users can draw colored lines in order to connect colored dots and thus draw a sketch. In this application, chords are used for selecting different colors. Chords in combination to personal windows found to be a good technique so as to understand individual actions on the screen. Moreover, we have developed a multiuser word construction application in which ChordiAction is used in order to differentiate user actions (change background letters, create a word etc.)

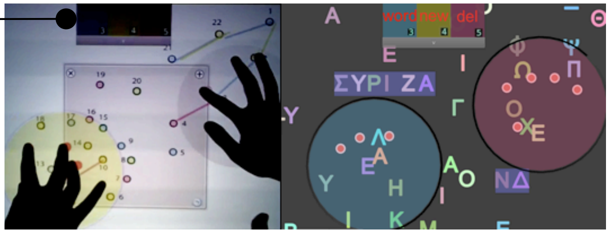


Figure 6: Snapshots of applications using ChordiAction.

Conclusion and Future Work

Considering the possibilities that larger MT surfaces present for collocated multi-user interaction, we presented ChordiAction, which is a toolkit developed as an open-source and ready to use platform that can be easily embedded in MT applications. The need for doing different actions depending on users' needs in the same multi-user application is also apparent. For example, in a scalable multi-user photo editing application one user is changing the brightness of a photo while another is changing the contrast of another. ChordiAction was designed to provide both services.

However, the development of software user interfaces that are rendered on devices with different input and output than those of the development workstation remains an open research issue. Additionally, the need for tools that are going to deal these issues emerges and toolkit developers should be aware of them, so as to design new development tools or improve existing ones for multi-touch devices. Our system was developed for larger vertical surfaces (Figure 7) and we plan to conduct some experiments to evaluate its efficiency with multiple users and compare our technique to other contextual menus or pie menus on large scalable MT surfaces.

References

[1] Hansen T. E., Hourcade J. P., Virble M., Patali S., and Serra T. 2009. PyMT: A Post-WIMP Multi-Touch User Interface Toolkit. Interactive Tabletops and Surfaces. ITS'09. ACM Press, New York, NY, USA.

[2] Kammer D., Keck M., Freitag G., and Wacker M., 2010. Taxonomy and overview of multi-touch frameworks: Architecture, scope and features. Symposium on Engineering Interactive Computre Systems. EICS '10 Workshop on Engineering Patterns for Multi-Touch Interfaces.

[3] Leftheriotis, I., and Chorianopoulos, K., 2011. Multi-user Chorded Toolkit for Multi-touch Screens, Proceedings of EICS ' 11. ACM SIGCHI Symposium on Engineering Interactive Computing Systems, ACM Press. New York, NY, USA.

[4] Lepinski, G. J., Grossman, T., and Fitzmaurice, G., 2010. The Design and Evaluation of Multitouch Marking Menus, Conference of Human Factors in Computing Systems. CHI '10, ACM Press, New York, USA.

[5] Shen C., Vernier, F. D., Forlines C., and Ringel M., 2004. DiamondSpin:An Extensible Toolkit for Around-the-Table Interaction. Conference of Human Factors in Computing Systems. CHI'04. ACM Press, NY, USA.