

# Device Composition on Tabletop Computers

Master Thesis by Leo Sicard  
*lsi@itu.dk*

March 2012

## Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum. Typi non habent claritatem insitam; est usus legentis in iis qui facit eorum claritatem. Investigationes demonstraverunt lectores legere me lius quod ii legunt saepius. Claritas est etiam processus dynamicus, qui sequitur mutationem consuetudinum lectorum. Mirum est notare quam littera gothica, quam nunc putamus parum claram, anteposuerit litterarum formas humanitatis per seacula quarta decima et quinta decima. Eodem modo typi, qui nunc nobis videntur parum clari, fiant sollemnes in futurum.

## Acknowledgements

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum. Typi non habent claritatem insitam; est usus legentis in iis qui facit eorum claritatem. Investigationes demonstraverunt lectores legere me lius quod ii legunt saepius. Claritas est etiam processus dynamicus, qui sequitur mutationem consuetudinum lectorum. Mirum est notare quam littera gothica, quam nunc putamus parum claram, anteposuerit litterarum formas humanitatis per seacula quarta decima et quinta decima. Eodem modo typi, qui nunc nobis videntur parum clari, fiant sollemnes in futurum.

# Contents

<b>I</b>	<b>Introduction</b>	<b>3</b>
<b>II</b>	<b>Design</b>	<b>5</b>
<b>1</b>	<b>Preliminary user study</b>	<b>7</b>
1.1	Method . . . . .	7
<b>III</b>	<b>Implementation</b>	<b>11</b>
<b>IV</b>	<b>Evaluation</b>	<b>13</b>
<b>V</b>	<b>Conclusion</b>	<b>15</b>
<b>VI</b>	<b>Appendices</b>	<b>17</b>

# List of Figures

1.1	Interaction primitives. . . . .	9
1.2	Interaction strategies. . . . .	10

# Part I

## Introduction

Tabletops should be used as peripherals  
Improve user experience  
both mobile devices and tabletops are made useful.

# Part II

## Design



Application Design = how did we design the application  
Design principles, inspiration, methodology  
Design concerns for tabletops  
scenarios  
ranking of features  
set of basic application commands set of interaction strategies  
preliminary usability study: method result analysis  
Design choices  
concept of META manipulation

# Chapter 1

## Preliminary user study

Designing a software application is no easy task, and implies making crucial decisions. AppName offers an unprecedented service and introduces new types of interaction, and thus challenging design.

A well-designed product is a successful one. Usability and appeal are key elements towards the success of an application. The goal of this experiment is to gather knowledge directly from users to inform important design decisions.

### 1.1 Method

Six interaction primitives were identified. They refer to the most basic user interactions that the application should support.

1. *Dragging* the application window across the interactive surface.
2. *Rotating* the application window across the interactive surface.
3. *Resizing* the application window across the interactive surface.
4. *Minimizing* the application window, making it possible to restore it easily.
5. *Hiding* the content of the application window.
6. *Exiting* the application, thus closing the application window.

Specific interaction strategies emerged from researching the different possibilities to implement those basic interactions. Each strategy can be consistently implemented for all previously defined primitives.

1. *Action Tabs* are traditional buttons/tabs that implement functionalities.

2. The *Action Bar* can be compared to a virtual touchpad, it includes a manipulation area and buttons.
3. *Window Toggle* refers to using a switch to toggle the window between inactive and active states. In its inactive state, the window is made manipulable as a common digital picture.
4. The *Active Border* is a digital frame around the application window used for manipulation.
5. *Active Corners* is a strategy similar to Active Border, with the difference that the border's corners implement specific functionalities.
6. *Other* regroups suggestions that do not fit with any specific strategy.

Figure 1.1: Interaction primitives.

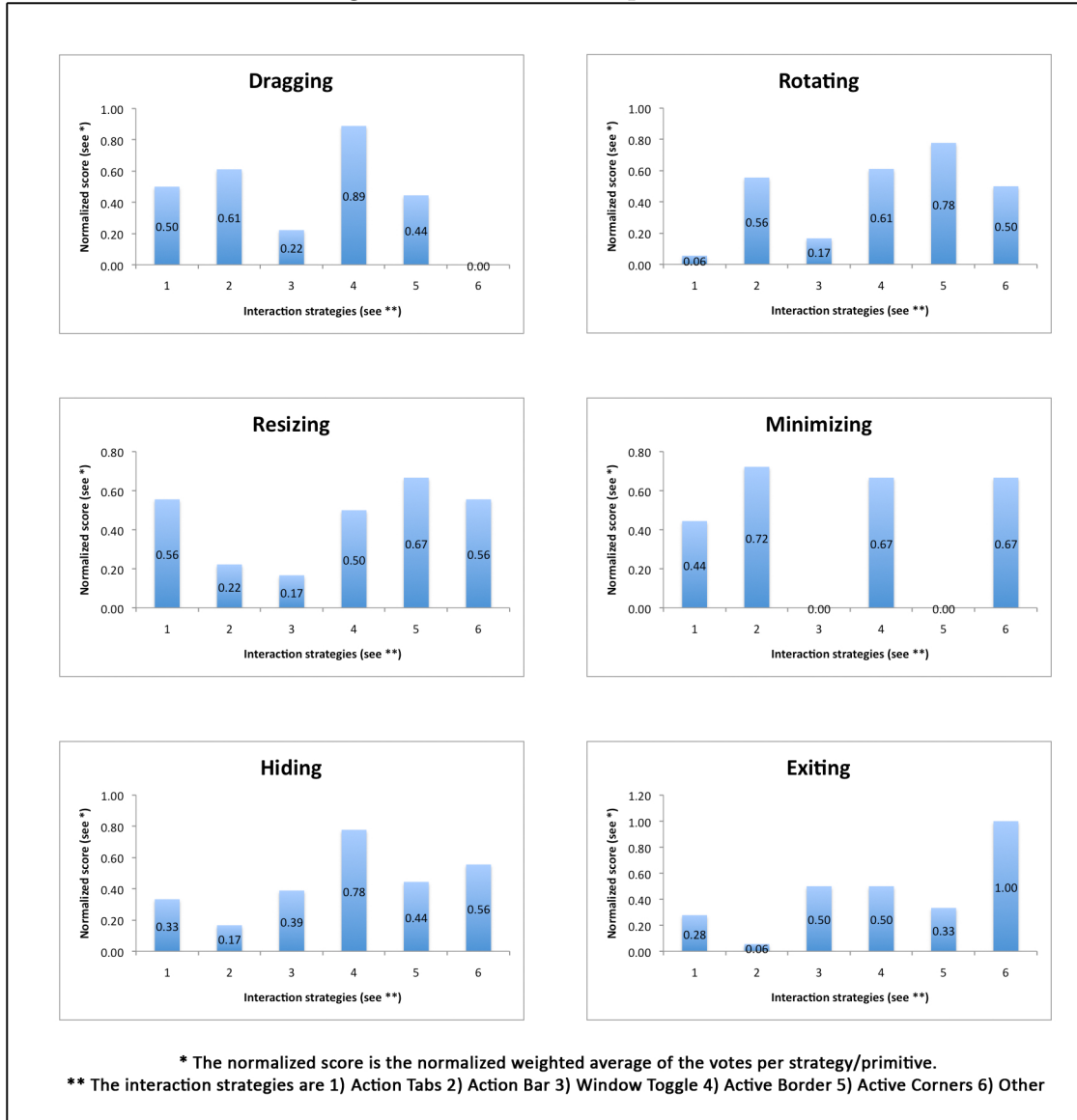
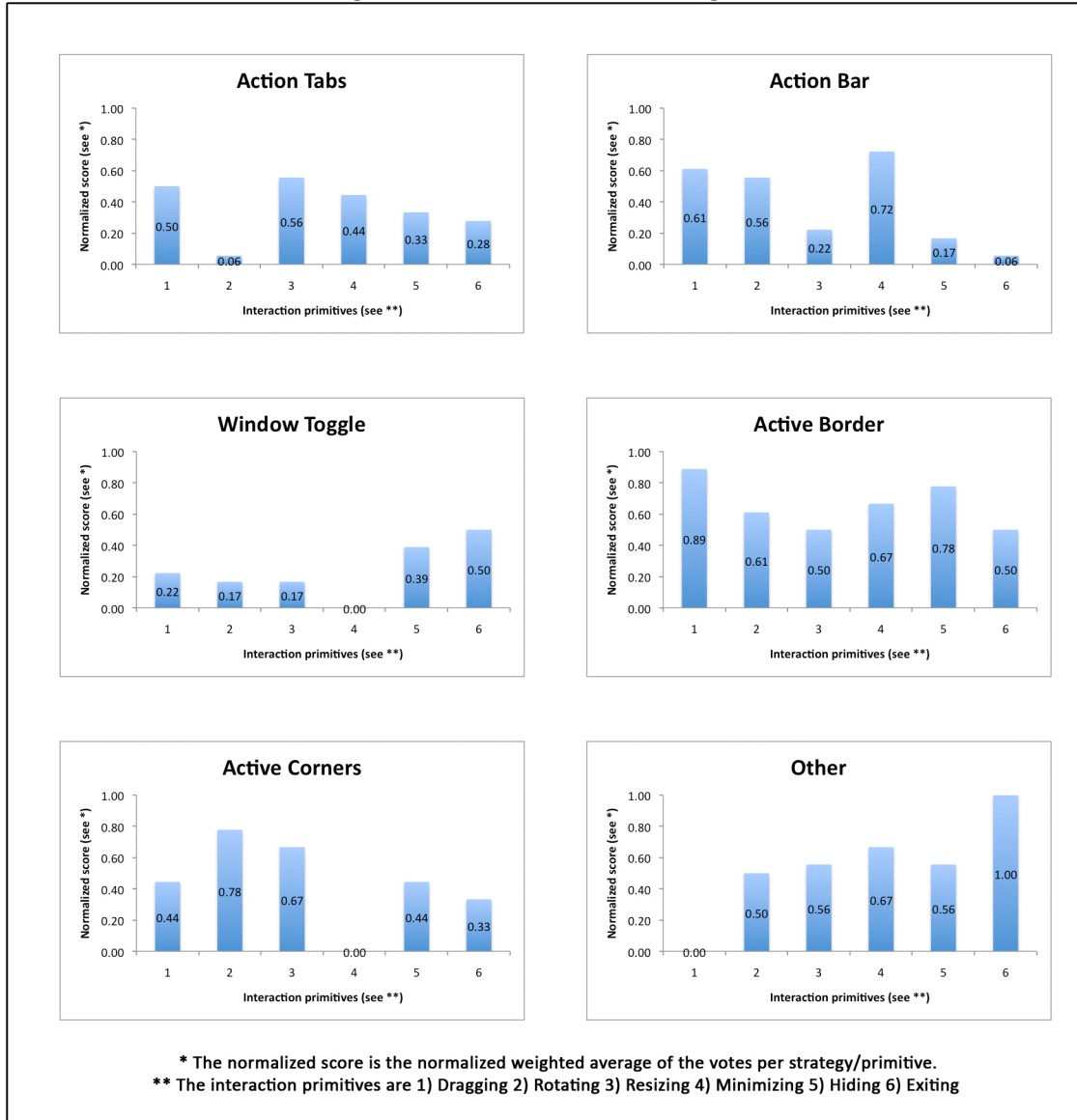


Figure 1.2: Interaction strategies.



# **Part III**

## **Implementation**

refer to problem formulation:

Following is an open list of problems that we will address in order to achieve device composition by means of implicit interaction.

1. *Setup*: How is a device enabled for integrating with a tabletop? The setup should be simple, to be performed only once by non-technical users. An initial survey of possible solutions points towards the use of tagging mechanisms and/or camera-based object recognition.
2. *Discovery*: How do the tabletop and the device discover and communicate with each other? How do we solve the issues of discovery, handshake, network connectivity, and encryption mechanisms to ensure privacy?
3. *UI transfer*: Given the computational constraints of mobile devices, how can the UI transfer be efficiently implemented so as to support native applications and guarantee a seamless user experience?
4. *Input*: How can the users interact with their applications on the tabletop (touch and other peripherals)?
5. *Interaction Design*: What means of interaction are best-fitted for the tabletop-based systems that we propose to develop? How can we best adapt to public/private uses and single/multiple users? How can we take advantage of the larger interaction surface?

1) setup

2) discovery

–¿ how it is not new, what are the existing options, what would I recommend in this context. Discussion. How did I solve it and why.

3) vision-based device tracking detection options: iPhone App, Tag, camera based

4) UI transfer (I/O approach) - technology issues (slow Veency)

5) Interaction design

6) generic implementation attempt include android phone

# **Part IV**

## **Evaluation**



separate implementations:

- 1) users best
- 2) designers best
- 3) mix

experiment to compare and discuss, use a timer

# **Part V**

## **Conclusion**

conclusion

# Part VI

## Appendices