RWTH AACHEN UNIVERSITY

HaptiCase: Tactile Cues on the Back of Devices for Eyes-free Absolute Indirect Tapping Tasks

Master's Thesis at the
Media Computing Group
Prof. Dr. Jan Borchers
Computer Science Department
RWTH Aachen University

by
Christian Cherek

Thesis advisor:
Prof. Dr. Jan Borchers

Second examiner:
Prof. Dr. Jochen Müsseler

Registration date: 19.08.2014
Submission date: 06.11.2014

I hereby declare that I have created this work completely on my own and used no other sources or tools than the ones listed, and that I have marked any citations accordingly.

Hiermit versichere ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Aachen, November 2014
Christian Cherek

Contents

List of Figures

List of Tables

Abstract

In situations where the graphical user interface on a smartphone is mirrored to a distant screen, it is difficult to hit on screen targets reliable. Users constantly have to switch the locus of attention between the content on the distant screen and the input device in their hands. We present *HaptiCase*, an interaction technique that provides tactile feed-forward to let users orientate on the screen dimensions, before interacting with the smartphone. Unlike other solutions our tactile landmarks are placed on the back of the device, and thereby still allow smooth gestures on the frontal touchscreen. Users move their fingers along the tactile guidance and perform a proprioceptive pinching gesture to transfer the location to the touchscreen at the front. We conducted a set of user studies to compare different landmark designs with a regular landmark-free smartphone. Users accuracy significantly increases when users were able to use *HaptiCase* compared to not having tactile guidance. In our study, users with *HaptiCase* hit 70.1% of the 15 mm targets compared to only 57.1% without tactile landmarks. A second study on the influence of tactile landmarks when looking at the input device, showed no significant effect of tactile landmarks on the performance. *HaptiCase* can be built as low cost prototypes, without the need to change the hardware or software of the smartphone.

Überblick

Wenn Benutzer ihr Smartphone als Eingabegerät für die Interaktion mit einem entfernten Bildschirm benutzen, kommt es oft zu Fehlern oder Falscheingaben. Das liegt daran, dass Nutzer ihre Aufmerksamkeit immer wieder zwischen der grafischen Benutzeroberfläche auf dem Bildschirm und dem Telefon in ihrer Hand aufteilen müssen. In dieser Master Arbeit präsentieren wir eine interaktions Technik, die takil erfühlbare Orientierungspunkte nutzt, um es Nutzern zu ermöglichen sich zu orientieren, bevor sie mit dem Touchscreen interagieren. Die Orientierungspunkte unserer Prototypen sind auf der Rückseite des Telefons, daher kann die Interaktion mit der Vorderseite immernoch wie gewohnt stattfinden. Nutzer können die Hilfestellungen auf der Rückseite nutzen, um sich zum Beispiel mit dem Zeigefinger zu orientieren, und anschließend mittels einer Kneifbewegung das entsprechende Ziel auf der Vorderseite mit dem Daumen auswählen. Unsere Nutzerstudien haben gezeigt, dass Nutzer mit *HaptiCase* eine signifikant höhere Genauigkeit erreichen als ohne taktile Hilfestellung. In einer zweiten Studie wurde gezeigt, dass unsere Orientierungspunkte keinen signifikanten Einfluss auf die Interaktion haben, wenn Nutzer die Möglichkeit haben das Eingabegerät zu sehen. *HaptiCase* wird in einem Lasercutter hergestellt, ansonsten benötigt es keine Modifikation der Hard- oder Software.

Acknowledgements

I would like to thank all testers and friends who participated in the user studies conducted for this thesis. Your willingness to help me finishing my studies was overwhelming. The feedback we collected during our user studies represents an important part of this thesis, thank your for that.

Secondly I would like to thank Prof. Dr. Jan Borchers and Prof. Dr. Jochen Müsseler. You offered valuable support and insight whenever I asked for it, the working environment you created motivated me constantly to improve further.

Thanks to the whole Media Computing Group Staff, whenever one of you dropped by, I got valuable suggestions and help from you.

A big thanks to Christian Corsten, M. Sc., my supervisor, it was a pleasure to work together with you on this topic. Your critique and suggestions were always constructive. I had few supervisors during my studies, who invested so much effort as you did for this. I really do look forward to collaborate with you in future projects.

I have to thank my friends and family for being very forgiving during the last months, I know I did not spend much time with you, thank you for supporting me and cheering for me.

Finally, I would like to thank you Lara, you are the one person in my life, without you all of this might not be possible. You give me strength to carry on, when everything seems to break down. You are the one person who would never let me down.

Conventions

Throughout this thesis we use the following conventions.

Text conventions

The whole thesis is written in American English. For reasons of politeness, unidentified third persons are described in female form. The first person is written in plural form.

Chapter 1

Introduction

Though everyday smartphone interaction typically happens at the touchscreen of the device, there are situations, where users interact with a smartphone, but they do not want to look at it. For example when the running application is mirrored to a distant screen , using techniques like Apples Air Play. Also more and more applications use this technique to utilize the smartphone as a controller for games. Figure 1.1 shows an example of a memory application mirrored to a distant screen.

There are situations, where looking at the phone is undesired.

Moreover, eyes-free interaction does not only happen when the graphical user interface is moved to a distant screen. Many TV sets come with applications to remote control the TV, or radio controlled toys use the phone sensors as input for their steering. In certain circumstances everyday smartphone interaction is prohibited or socially unaccepted. For example while driving a car, incoming phone calls can not be accepted, since the users attention is needed on the traffic. There are applications, who allow users to decline an incoming call and send predefined text messages like *"I am not able to take the call, I'll call you as soon as possible"*. This interaction however needs to be done eyes free as well.

Typical everyday smartphone interaction is *absolute* and *direct*. A target on the screen is activated by tapping at a certain position, and lifting the finger again. The position is absolutely mapped to the graphical user interface (GUI).

Everyday smartphone interaction is absolutely and directly.

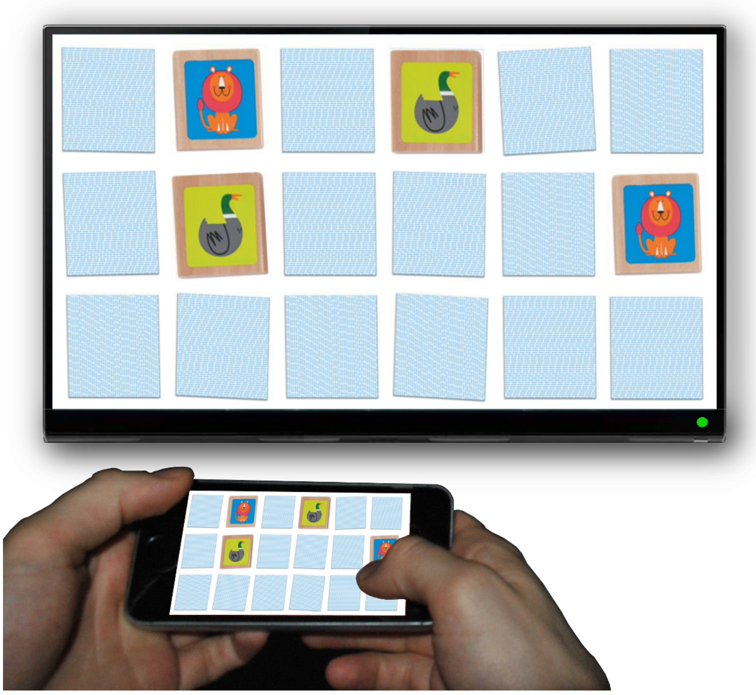

Figure 1.1: A memory game mirrored to a distant screen

Since the touch sensor and the smartphone screen are collocated, users get the impression of directly interacting with the interface.

In eyes-free interaction, user indirectly interact with the smartphone. This leads to errors.

However, when the interface is mirrored to a distant screen or the phone is accessed eyes-free, the impression of direct interaction does not hold anymore. Instead users *indirectly* interact with the GUI on the distant screen, or their mental image of the interface. This *absolute indirect* touch interaction is difficult, since users have to switch their locus of attention constantly between the input device and the distant screen. When trying to aim for targets without looking at the screen, users might miss the desired target, or hit the wrong one.

HaptiCase alleviates this problem.

To address this problem, we present *HaptiCase*, an interaction technique aiming to diminish the need of visual contact to the input device, by offering tactile tracking instead. This is achieved by adding tactile landmarks to the back of the device. Along these landmarks, users can orientate

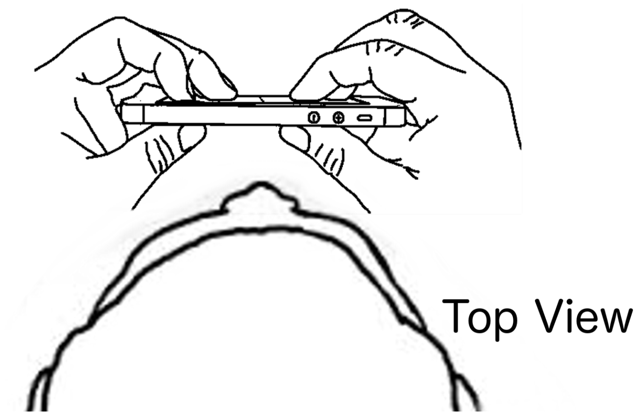

Figure 1.2: Top view how users interact with *HaptiCase*

themselves on the devices spacial dimensions. Figure 1.2 shows the interaction with *HaptiCase* from top view. The user orientates herself with the index finger, before interacting with the touchscreen. Since the orientation happens before users interact with the phone, *HaptiCase* can reduce targeting error in eyes-free tapping tasks. If the position is found, users perform a pinching gesture to transform the location from the back to the front. This pinching gesture is based on human proprioception, and therefore can be performed eyes-free.

Tactile landmarks offer guidance.

Our prototypes are built in a laser cutter, and glued to commonly available phone protection cases. *laser cutter* Therefore, *HaptiCase* is a low cost solution for targeting problems in eyes-free tapping tasks. Furthermore, *HaptiCase* does not need any other hardware or software changes of the input device.

HaptiCase is a low cost solution

Main contributions of this thesis will be, the *HaptiCase* interaction concept, the experiments we conducted to measure the performance in eyes-free tapping tasks, and a set of implications for designers, who want to utilize *HaptiCase*. The *HaptiCase* interaction concept will be explained in detail in chapter 2. Chapter 4 will explain our designs and the manufacturing of the low cost *HaptiCase* prototypes. We performed a set of user studies, to evaluate if and how *HaptiCase* influences users accuracy in eyes free tapping tasks. Chapter 5 gives detailed descriptions of our user studies, results, and conclusions we drew. To give a vision, how *HaptiCase* could be designed for everyday interaction we collected a set of implications in chapter 6. There we will also name suggestions for application designers, to create an application that works well with *HaptiCase*.

Chapter 2

Interaction Technique

2.1 The HaptiCase Interaction Technique

Interaction with mobile touchscreens follows a Two-State Model, as described by Buxton [1990](Fig. 2.1). While the user is pointing with the finger to the screen, to select an on screen target, *State 0* is maintained. During this state, the system does not know what is being pointed at right now, therefore Buxton called this tracking *passive*. When the finger touches the screen, *State 2* is entered. As long as the finger stays in contact with the screen, the system maintains *State 2*: *Selection*. When the finger *releases contact*, *State 0* is re-entered. To hit targets, users typically touch the screen and immediately lift the finger again. This interaction is called *tapping*. The *Selection* is immediately fired, when the users finger touches the screen. Therefore, users have to hit a target on the first attempt, since correcting the finger position before lifting is not possible after entering *State 2*. To hit an on-screen target correctly, users have to look at their fingertip, aim for the target, touch the screen, and lift the finger up again. Touching another adjacent target might trigger an undesired event. Therefore, visual contact is necessary to interact correctly with a mobile touch device.

Buxton et al. described interaction with state charts.

From the systems point of view, touching is the only interaction.

Buxton's model is designed to describe interaction from the perspective of the input device. Therefore, the model does not distinguish between a state, where no interaction

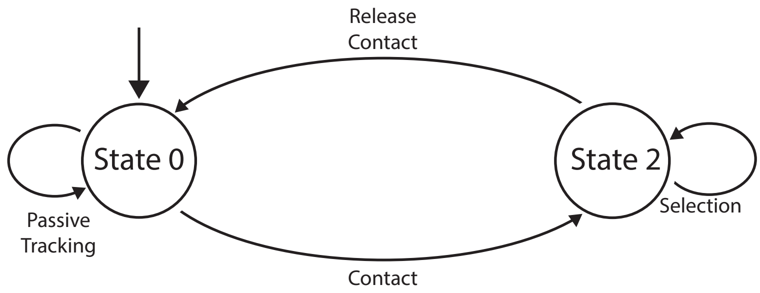

Figure 2.1: The State 0-2 Transitions as presented by Buxton [1990]

happens at all, and the visual targeting performed by the user, while aiming for on screen targets. From the systems perspective, there is no difference in not being interacted with at all, or used for visual targeting. We extended Buxton's *State 0–2 Transitions* model with a *Visual Targeting* state (Fig 2.2).

From the users point of view, interaction starts when aiming for on-screen targets.

Stating in *State 0*, the finger is out of range of the input device. When the user moves her finger above the touchscreen, the *Visual Targeting* state is entered. In this state, the user aims for on screen targets, looking at her fingertip to hit the correct target. The system does not know about the state change from *Out of Range* to *Visual Targeting*. Only when the user touches the screen, the system recognizes the touch, and fires the corresponding touch event. This is represented with *State 2: Selection*.

This strategy is used for tapping tasks on mobile devices, as long as the user is able to look at her finger above the screen of the device. However, when not beeing able to look at the input device, users cannot change from *Out of Range* into the *Visual Targeting* state. Therefore, tapping accuracy will likely be heavily reduced, which leads to missing on screen targets as well as hitting wrong targets instead.

With *HaptiCase*, we introduce a *Tactile Targeting* state to enable accurate tapping for situations in which users do

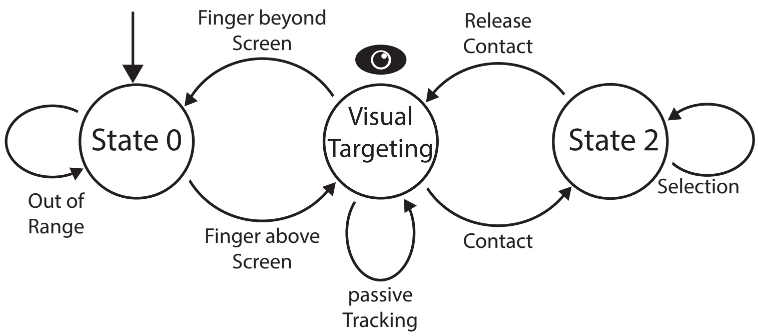

Figure 2.2: Our extended version to visualize the visual targeting of the user.

not have eye contact with the touchscreen. *HaptiCase* provides tactile landmarks on the back of the input device, and thereby offers guidance on the devices spacial dimensions.

HaptiCase introduces the Tactile Targeting State.

While holding the device, users explore the tactile landmarks by moving their fingers on the back. This state corresponds to the *Tactile Targeting* state in Figure 2.3.

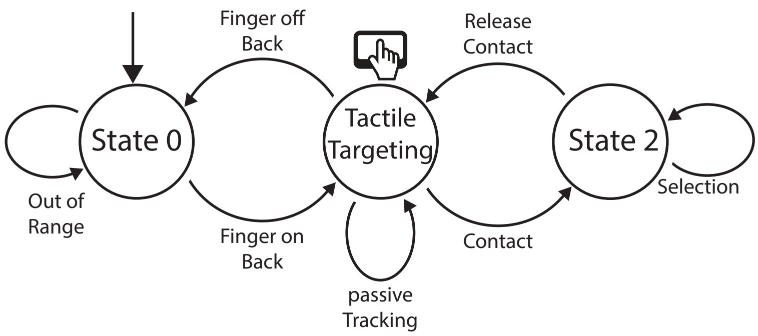

Figure 2.3: The extended version of Buxtons state chart with a tactile targeting state when a users finger is placed on the back

When the user identifies a tactile landmark close to the position of the on screen target she is interested in, she pinches her thumb to the finger on the back. This pinching gesture (Fig. 2.4) can be performed accurately and eyes-free, since

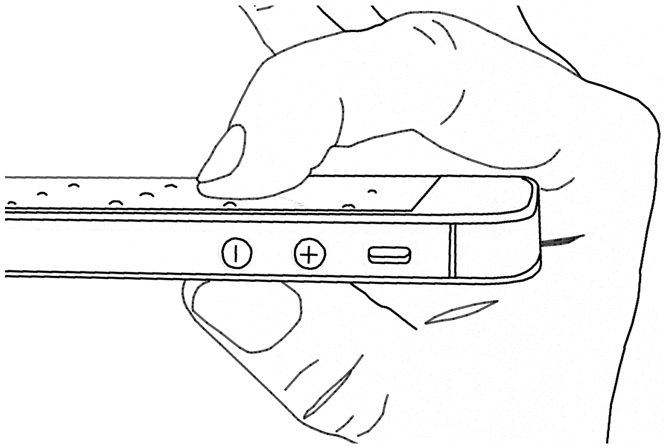

Figure 2.4: a pinching gesture is performed to transform a location on the back of the phone to the front, to touch an on-screen target.

After tactile targeting, a simple pinch selects an on screen target.

it is based on human proprioception. With pinching, the user transforms the location on the back to the front, and thereby activates the on screen target.

HaptiCase is meant to be used eyes-free.

Typical use cases for *HaptiCase* are tapping-based smartphone applications, that are accessed eyes-free. An interactive presentation with on-screen targets to play movies or trigger animations could be such a scenario. To interact with the interactive targets while still looking at the audience, the presenter utilizes her *HaptiCase*-equipped smartphone as presentation tool. The tactile landmarks of *HaptiCase* are placed on the back of the device, and therefore still allow smooth gestures on the front, like flipping slides during the presentation.

Chapter 3

Related work

HaptiCase interleaves with multiple research fields, to give an overview on related research we will name three different fields that are closely related to our interaction technique.

HaptiCase is related to Absolute Indirect Touch Interaction, Feed-Forward techniques, and Back-Of-Device-Interaction

Absolute indirect touch interaction focuses on interaction with distant screens. Related work in this field often searches for more efficient ways to control graphical user interfaces on a distant screen, for example by combining absolute and relative manipulation of a cursor. Basic research on touch accuracy in eyes-free interaction as well as recommendations on the aspect ratio of input and output device belongs to this field. **Feed-forward** or a-priori feedback is the second topic related to *HaptiCase*. This field focuses on system design, that allows users to get a image of the system state, before a selection is performed. Some feed-forward prototypes read out on screen targets, to enable visual impaired people to use touchscreens. These prototypes often have to come up with new interaction modes, to enable screen exploring without manipulation of the graphical user interface. An example would be a lift-and-tap technique, where on screen targets are selected by moving the finger to the target, than lifting the finger, and tapping to perform the selection.

Back-of-device interaction is closely related to *HaptiCase*, since our landmarks are placed on the back of devices. Research in this area focuses on interaction with the surface, that is facing away from the user. For example by adding a camera to the back of a touch device, and thereby using the back as input surface. Research is this area often is motivated by the fat finger problem as described by Siek et al. [2005].

3.1 Absolute Indirect Touch

Absolute indirect touch interaction is a research field, that is approached from different angles. The ARC-Pad by McCallum and Irani [2009] is a trackpad, that offers users absolute indirect pointing to position a cursor on a distant screen. When the cursor is roughly at the correct position, users can use swipe gestures to enable relative pointing, that allows more precise cursor placement. This is especially needed, when the screen is a lot bigger than the input device. Users were faster using ARC-Pad, compared to build in relative positioning, without reducing the pointing accuracy. Building upon the work of ARC-Pad, Nancel et al. [2013] added a two-finger technique for absolute interaction. However, for typical interaction with a smartphone ARC-Pad is not suitable, since users places a cursor instead of directly manipulating the buttons on the screen. For eyes-free interaction ARC-Pad is a well suited idea, since selection with a cursor does not create the need to look at the input device.

Gilliot et al. [2014] investigated different input conditions for absolute indirect touch tasks. Users performed tapping tasks with their index finger on different sized input screens, and with different output screen sizes. The influence of eye contact and target size had been evaluated as well. An important conclusion Gilliot et al. [2014] named, is that designers should keep the aspect ratio of input and output device the same. Differences in size however can easily be compensated by users. When looking at the input device, targets should be $\geq$ 16.8 mm, if the user is not able to see the input device $\geq$ 23.0 mm. The trackpad Gilliot et al. [2014] used as input device, was placed steadily on a desk. They also did not offer tactile guidance, excerpt the device dimensions. *HaptiCase* builds upon this research, by using a mobile input surface, as well as tactile landmarks to offer orientation in eyes-free tapping.

Pietroszek and Lank reported a 62 px vs. 34 px respectively 3.07 mm vs. 1.68 mm targeting error when using a touchscreen either without mirroring of the interface on the input device, or with targets visible on the screen. The touchscreen they used had 48×36 mm^2. Unfortunately, their results were means over very differing target sizes from 2.4–12.0 mm, and they did not state which finger was used. In all conditions, users were able to look at the input device.

An interesting approach to absolute (indirect) touch is The Imaginary Phone by Gustafson et al. [2011]. They used the human hand as surface for touch input for a imaginary user interface. They rely on users spacial memory, such as the home screen of their smartphone to enable interaction. When touching the palm of the non-dominant hand, the system reads out the target, and with an additional tapping users select a target.

Though the authors state the ability to see the hand is an important for targeting accuracy, users were able to hit 17.7 mm targets on the imaginary GUI reliable. In a follow up work about imaginary interfaces, Gustafson et al. [2013] found, that tactile cues can be helpful for targeting accuracy. However, to sense the input on the palm, users need to equip their hand with technical instruments.

HaptiCase might be helpful in similar situations, a memorized graphical interface could be manipulated just with tactile guidance from *HaptiCase*. This would enable in-pocket interaction, or accessing functionality at memorized locations eyes-free.

Whereas the research described so far considered mobile touchscreens, Voelker et al. [2013] evaluated different state switching methods for absolute indirect dragging tasks from a 27” touchscreen to an equivalent-sized display. Based on their evaluation, the authors suggest to use a lift-and-tap technique: The user moves her finger across the input screen, holding the finger on the surface, and receives visual feedback on the display. When a target is crossed, it can be activated, by lifting the finger and tap again. Since this technique requires mode switching between exploring and engaging, it is not possible to immediately select a target. Furthermore, the authors state that, using lift-and-tap, “two nearby touches may trade places, especially in a small device” like a smartphone Voelker et al. [2013].

3.2 Feed-Forward for Eyes-Free Touch Interaction

As Gilliot et al. [2014] stated, it is difficult to interact with a GUI when not being able to look at it. This is especially true, when the GUI is on a plain surface touchscreen without any possibility to orientate with other senses. Research in the feed-forward field often focuses on enabling visual impaired people to use touchscreens. Since visual impaired cannot use their visual sense to interact with the screen, often the aural or tactile sense is used to offer feed-forward.

Typical touchscreens do not offer a exploration state, since touching the screen directly starts the interaction. This makes it difficult for visual impaired to interact with touchscreens. Therefore many approaches to overcome this problem, introduce a new state to explore the interface, before interacting with it. The lift-and-tap technique by Voelker et al. [2013], is an example for such a *touch-and-explore* interface. During the exploration phase, users slide their finger over the display and the system gives feedback on the targets beneath the finger. Especially for visual impaired people this often is done via audio output. VoiceOver in Apples iOS is an example for that, Vanderheiden [1996] presented the “Talking Fingertip Technique”. Zhao et al. [2007], and Kane et al. [2008] offer similar ideas. For UIs with only few targets, Kane et al. [2011] presented a technique to translate touch targets to the edge of the screen. However, this technique only works if the screen has only few targets.

Auditive feed-forward has other drawbacks as well. Reading out a on screen target requires time, therefore the interaction speed is limited. Exploring an unfamiliar interface might require a lot of time, since user cannot get an immediate overview. Additionally auditive feedback is not always suited, in crowded places users cannot use their phone privately. Also in gaming scenarios, where audio is already used to enhance the user experienced are not suitable for additional audio feedback.

We are not the first who came up with the idea, to use the tactile sense to explore a graphical user interface. Landau and Wells [2003] introduced The Tactile Talking Tablet, that uses relief paper and audio feedback to enable visual impaired people to use a touch device. Users were even able to interact with the relief buttons to control the user interface.

Tactile overlays for a touchscreen were presented, by Kincaid [2012], El-Glaly et al. [2013], and Kane et al. [2013]. These overlays are transparent be able to see through, with cutouts for locations of on screen targets. However, most of these are application-specific, that means each user interface needs another tactile overlay. Touchplates by Kane et al. [2013] are more generic, but are created for non mobile touch screens.

Guerreiro et al. [2011] added tactile grids to touchscreens for smartphones and tablets. Visual impaired people can orientate along the grid, and tap adjacent areas to select a target. Their results showed a significant affect on users performance, additionally users stated to like the grid a lot. Unfortunately, this also only allows very few touch targets, also performing gestures on the device is hindered by the tactile grid. The tactile cues *HaptiCase* offers are on the back of the device, and therefore still allow smooth gestures on the touchscreen.

Frey et al. [2011] introduced a possibility to type Braille on a touchscreen, also offering tactile landmarks to orientate. Users hold the smartphone with the screen facing away from them and type with index, middle and ring fingers. On the side facing towards the user a tactile grid was placed, this grid could be used to hold the device with the thumbs, and to estimate the button positions on the touchscreen. However, users sated this posture to be really fatigue to hold, and the tactile guidance is difficult to feel with the thumbs. Also Frey et al. [2011] did not investigate their prototype with tactile guidance. *HaptiCase* offers tactile landmarks for a variety of applications, and we performed a detailed analysis on the influence on tapping accuracy.

Another idea to help users orientating on the side of the device was presented by Pielot et al. [2012]. They used a prototype called PocketMenu to offer users one dimensional guidance along the side of the smartphone. This required them to place all targets along a vertical alignment, since the landmarks are only placed in one dimension. *HaptiCase*s tactile landmarks are two dimensional, just as the user interface, therefore they offer guidance for a broader variety of graphical interfaces.

Buzzi et al. [2013] marked certain UI segments, such as the menu bar or similar lines in a smartphone interface, with haptic spheres at the side of the phone. Moving the finger along the side, reads out targets located within a segment. The authors presented the concept based on the idea to render the tactile cues dynamically corresponding to the current interface. This dynamic rendering was not provided by Buzzi et al. [2013], instead they built fitting prototypes for their experiments.

In the field of dynamically rendered tactile feedback, Yatani and Truong [2009] presented an approach to give feedback through vibration patterns. Users had to learn a set of vocabularies to identify different touch targets, to be able to work with their prototype, called SemFeel. GraVVITAS Goncu and Marriott [2011] uses vibration motors attached to the user's hand to haptically "visualize" graphic elements. This allows users to sense basic shapes on a touchscreen, unfortunately a complex prototype needs to be used, since the vibration motors are attached to a glove.

TeslaTouch Bau et al. [2010] used electrovibration to create tactile feedback for touch surfaces, however their feedback is more suitable for gestures or moving the finger, than tapping. Ultra-Tangibles Carter et al. [2013] work with Ultra sound, to move physical objects. VacuumTouch Hachisu and Fukumoto [2014], and FingerFlux Weiss et al. [2011] attract users fingers via air suction, or magnetic pull. MudPad Jansen et al. [2010] provides localized haptic feedback using a magnetic fluid and an array of electromagnets. Programmable Friction Levesque et al. [2011] used a Large Area Tactile Pattern Display to program tactile feedback. All these solutions however, share a large scale factor, and need complex hardware setups. *HaptiCase* is used on mobile touchscreens, and does not need a complex setup to work well.

3.3 Back-of-Device Interaction

HaptiCase introduces tactile landmarks on the back of the device to offer orientation on the phone dimensions. Therefore a important part of our interaction happens on the back of the device.

Wigdor et al. [2006] presented a two sided direct touch tabletop. They investigated different use cases and a design space for the interaction on the bottom side of the table. One possibility they mentioned was absolute indirect interaction with a distant screen. They performed a user study and found, interaction on the back to be significantly less accurate, compared to the top surface. Wigdor et al. [2006] did not offer any tactile guidance to help users find a target on the invisible surface.

To investigate human performance on the front or back of a device, Wobbrock et al. [2008] performed a user study, where users had to drag targets over the screen. These tasks were performed with either the index finger, or the thumb, while the device was held in one or two hands. The results indicated, that the index finger works better for this interaction than the thumb, on both, the front and the back side. During this study, users were allowed to look at their hand the whole time. Though dragging is an interesting task in touch interaction, for absolute indirect interaction, such as remote controlling a television set, tapping is more relevant.

To circumvent the fat finger problem as described by Siek et al. [2005], Wigdor et al. [2007] added a camera to a mobile touch screen, to be able to capture finger input on the back of the device. The shape of the finger on the back is displayed on the front, and thereby offering the illusion to look through the device. Building upon this, Baudisch and Chu [2009] investigated interaction on the back of the device for a small touch screen. Their prototype used a capacitive touchpad in a 2.4″ device. The mapping from back to front was absolute, and the fingers on the back were visualized lying behind on screen targets. Their study found $12.2 \times 12.2\ mm^2$ targets to be the minimal size for users to hit targets with at least 90% accuracy. However Baudisch and Chu [2009] focused on very small touch screen devices, and did not address everyday smartphone interaction.

Chapter 4

Design and Fabrication

4.1 Idea and Design

As described in chapter 2, we added tactile landmarks to the back of a mobile touchscreen to enable feed-forward guidance on the device dimensions. We envisioned a prototype that enables users to use the smartphone accurately, although they do not look at the input device. Additionally, we wanted *HaptiCase* to be a low cost solution, without the need to change the hardware and the software of the smartphone. Instead of creating a tactile explorable copy of an interface on the back, *HaptiCase* was designed to offer tactile guidance independently from the current user interface.

HaptiCase offers feed-forward guidance.

Originally, we created eight different tactile landmark designs for *HaptiCase*. Figure 4.1 shows these designs including a landmark-free case design, that was used as baseline in our experiments. The *Base* design together with the smartphone has the same thickness as other case designs and the smartphone. Therefore, the feel of the surface in *Base* is the same as if we simply used another design, but without the landmarks.

We created eight different designs.

The *HaptiCase* design called *Frame*, has a frame with tactile lines corresponding to the edges of the touchscreen at the front and measures 90.25×51.6 mm^2 in size. Figure 4.1

Frame offers basic guidance.

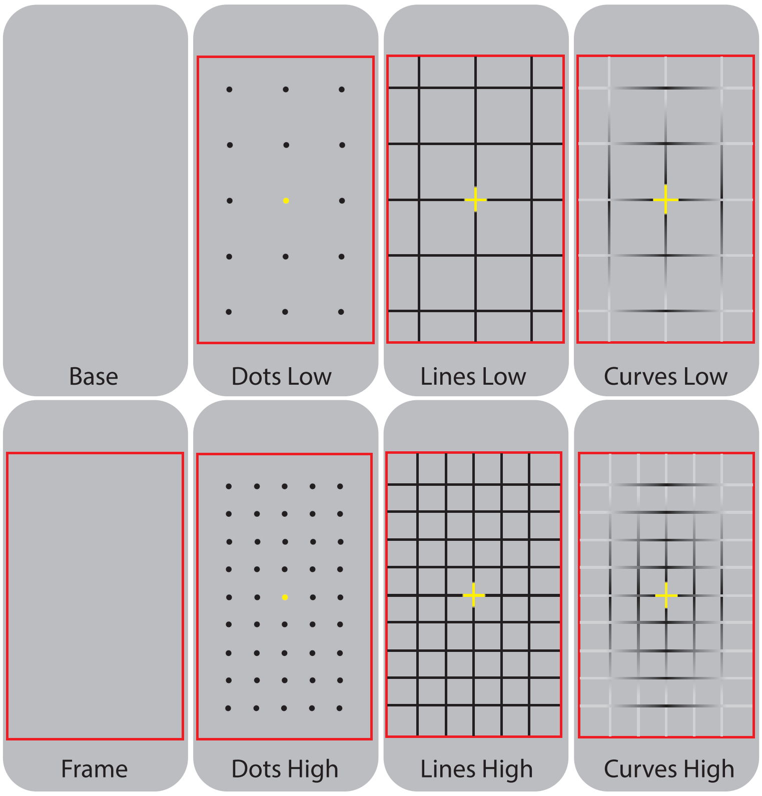

Figure 4.1: Our eight different *HaptiCase* designs: 1. landmark-free *Base*, 2. *Frame* with the outline of the screen, 3. *DotsLow*, and 4. *DotsHigh* featuring small dots as landmarks, 5. *LinesLow*, and 6. *LinesHigh* with continuous landmarks, 7. *CurvesLow*, and 8. *CurvesHigh* with landmarks that change in height from the border to the center. Screen borders are highlighted red, screen center in yellow.

highlights the frame in red, the *Frame* is used in the other case designs as well.

Dots and Lines designs have more elaborate tactile cues.

We added small gradient dots within the screen borders to the *Dots Low* design. These dots are laid out on a 3 $\times$ 5 grid, with equal distances between adjacent dots. The second dot-based design, *Dots High*, has additional dots between each dot from the low resolution design. The higher resolution was designed to offer a tighter grid of orientation points.

However, the *Dot* designs can only give concrete reference points. This might be not sufficient for orientation tasks, that need to be performed precise. Therefore, we created two designs for *HaptiCase*, with continuous lines as guidance. Users can slide their finger along these lines to have a continuous reference. We built a *Lines Low* and a *Lines High* design. Each intersection of two lines thereby corresponds to a dot in our *Dots* designs.

Continuous landmarks, as used in our *Lines* designs, have the side effect, that user might not be able to distinguish between the screen frame and the landmarks within. This could lead to confusions, since users are not able to take a quick look, whether their fingers are placed on a landmark towards the center or not. To alleviate this problem, we came up with a landmark design that had continuous lines as used in the *Line* designs, but this time with changing heights of the landmarks. *Curves Low* and *Curves High* have tactile lines, with continuously changing height over the back. Towards the center, the lines in a *Curve* design have the same height as in a *Line* design. More towards the border, the height of the line is continuously reduced. We aimed to offer additional orientation through the height change. A user should be able to identify, whether she is on the screen border with continuous height, or on a landmark within the screen. When sliding the finger over a height changing landmark, users can feel, whether they are more at the center of the screen, or more at the border, where the line is comparatively low.

The most complex prototypes are the *Curves* designs

4.2 How to Build a *HaptiCase*

We created our *HaptiCase* prototypes with a Epilog Zing 6030 lasercutter, as it is provided in the fab(rication) lab(oratory) by the Media Computing Group[1] at RWTH Aachen University. The designs were sketched with Adobe Illustrator and afterwards 3D-engraved into 2mm acrylic sheets. Table 4.1 lists detailed information, how to set up the laser cutter to manufacture *HaptiCase*.

HaptiCase prototypes are built with a laser cutter.

[1]http://hci.rwth-aachen.de/fablab

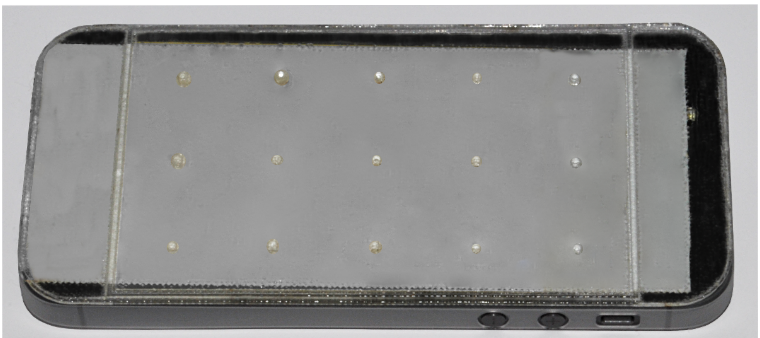

Figure 4.2: The back of an iPhone 5S with the attached *Dots Low* design of *HaptiCase*

The original thickness of the acrylic was 2 mm after engraving our designs, dots and lines were at maximum 0.45 mm raised above the rest of the material. The dots had a diameter of 2.5 mm and the lines were 2 mm wide. Engraving big areas in the acrylic left a rough surface, which could lead to confusions, since the rough surface might be misinterpreted as tactile landmarks. Therefore, we covered the rough areas with smooth plastic sheets, to flatten the area and allowing for undisturbed exploring.

Vector Marking Speed	20
Vector Marking Power	100
Vector Marking Frequency	5000

Table 4.1: Laser cutter settings to manufacture a *HaptiCase*

thin plastic sheets smoothen the feeling on rough areas.

This let to a smooth feeling while sliding a finger over the back of the case. Figures 4.3 and 4.2 show the individual parts before being glued together and a completed 3×5 *Dots Low* design attached to a iPhone 5S. The landmark-free *Base* design was engraved as every other design. That enabled us to use it as a landmark-free baseline for our experiments, still with the same dimensions as the ones with landmarks.

To attach the acrylic to the phone, we used 0.35 mm thin phone protection cases[2]. This way we could easily ex-

[2] http://chimpcase.de/

Figure 4.3: A *HaptiCase* prototype engraved into acrylic, and a phone protection case, as used for *HaptiCase*

change different *HaptiCase* designs during our studies. This is also practical in everyday life. For example, a user might use a universally designed *HaptiCase* throughout the day for eyes-free interaction with the phone interface, but at home, when playing a game at the television screen, she replaces her phone case to a *HapriCase* design specially created for this game.

glued to phone protection cases, the prototypes can be exchanged easily.

Chapter 5

Evaluation

We conducted a set of user studies, to evaluate the impact of *HaptiCase* on users tapping accuracy. We wanted to investigate, if *HaptiCase* can offer valuable guidance in eyes-free tapping tasks. Additionally we aimed to find out which landmark design might be most useful, what users think about our idea, and how the timing performance is influenced by *HaptiCase*. For preliminary insight on our design ideas and reasonable hand postures, we performed a preliminary study with ten users. We than conducted two users studies with 24 for the first and 12 users for the second experiment. During the first study we evaluated six different *HaptiCase* designs, investigating, which might be most useful. The second study focused on the influence of the visual sense. We wanted to know, how tactile targeting performs compared to visual targeting.

5.1 Preliminary Study on Ergonomic Experience

The goal of our preliminary study was to find out about reasonable postures to use *HaptiCase*. At first we envisioned *HaptiCase* to work with both, one-handed usage in portrait mode, and two hand usage in landscape mode. In one-handed portrait mode, users hold their device with a hand, using the thumb as input finger. Instead, in landscape mode both hands are used to hold the device, and both thumbs to hit on screen targets. As we wanted to know, if *HaptiCase* can be used in both orientations, we conducted a preliminary study on that. In addition, we let users perform some trials with the different *HaptiCase* designs, to get insight whether our design decisions were reasonable.

5.1.1 Apparatus and Task

As we wanted to let users perform an absolute indirect tapping task, we set up a desk with the study setup. Users used an iPhone 5S as input device, the target positions were presented on a distant screen. We will describe the study setup in more detail in 5.2.2. We attached a complexity reduced *HaptiCase* design to the input device. As this experiment did not aim to find out about tapping accuracy, we decided to create this additional design. This design had a set of five dots at the four corners and the center of the screen, as well as the frame outline of the touchscreen (Fig. 5.1). We let all users perform the task once in portrait mode, with only one hand to hold and tap the device, and once in landscape mode holding the phone in both hands typing with both thumbs. To avoid learning effects we let half of the users start in portrait mode and the other half in landscape.

The task was to hit a set of 15 targets, that were placed centered on the position of the five landmarks at the corners and the center. Comparing target sizes from related work (Chap. 3.1), we decided on a target size of 10×10 mm^2. This size also roughly corresponds to a button in Apples iOS. We measured the success rate of users, and collected oral feedback on how using *HaptiCase* felt for our users.

Additionally, we let some users perform a set of test trials with all eight *HaptiCase* designs. We asked users for oral feedback, whether they could utilize *HaptiCase* to tap more accurate. We also asked whether users felt comfortable with *HaptiCase*, or if the landmarks were more confusing than helping. Since we were merely interested in

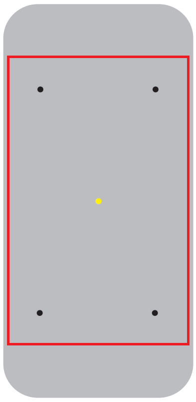

Figure 5.1: The HaptiCase design we used in our pre-study. The screen border is highlighted in red, the center in yellow.

users experience, we did not evaluate the accuracy results of these preliminary studies in detail.

5.1.2 Results

An evaluation of user hit rate during our first experiments showed, that 10 mm targets are not suitable for the eyes-free tapping task. Less than 45% of the targets were hit at the first attempt, though we reduced the set to only five different positions. Consequently we increased target size for follow up experiments to 15×15 mm^2.

The oral feedback we collected during the preliminary experiments gave valuable insight about *HaptiCase* designs, and reasonable hand postures.

HaptiCase Designs. Regarding the different *HaptiCase* designs, our users stated that the *Curve* designs were to confusing to be useful. First users did not recognize the continuously changing height of the landmarks. Only when sliding the finger towards the border of the screen, a different height was recognized. Since users did not expect this gap, they were confused about it, and could not compensate for that during the task. Therefore, we decided to remove the *Curves Low* and *Curves High* designs from future experiments.

Posture Usage. While most users mentioned *HaptiCase* to be useful in landscape mode, portrait mode did show to be less suitable. In portrait mode, users are not able to reach the whole screen with their thumb, therefore they need to re-grasp the device frequently to reach all targets. Often users had to hold the device in a posture, where they were not able to explore the tactile landmarks any more, since index, middle- and ring finger were needed to balance the device in one hand. This, combined with the wish to reach the tactile landmarks led to an unstable positioning of the phone in users hands. Some users even accidentally dropped the phone during the experiment, or needed their second hand to steady the device in their hand again. Therefore we decided to only use landscape mode for further investigation.

5.2 Experiment 1 on Eyes-free Tapping Accuracy with HaptiCase

The first experiment aimed to investigate the influence of tactile landmarks on eyes-free absolute indirect tapping tasks. Users performed a set of tapping trials, to compare targeting accuracy with *HaptiCase* to the landmark-free *Base* design.

5.2.1 Hypothesis

For this experiment we formulated three hypotheses:

1. Tapping an on-screen target of 15 mm width with *HaptiCase* is more accurate than with the landmark-free *Base* design.

2. High resolution *HaptiCase* designs will increase users accuracy, compared to low resolution designs.

3. Targets towards the center will be hit less accurate than targets close to the screen borders.

5.2.2 Apparatus

During our experiment, users used an iPhone 5S with a 4″ screen (1136 $\times$ 640 px^2), and a device size of 123.8 $\times$ 58.6 $\times$ 7.6 mm^3. Since users were not able to look at the device, we left the screen blank. The study ran on a 2009 Mac Book Pro with a 30″ display attached to it. The resolution of the display was 2560 $\times$ 1600 pixels. To match the aspect ratio of input device and the screen, as recommended by Gilliot et al. [2014], we attached a custom made cardboard bezel with a 284 $\times$ 160 mm^2 cutout at the center.

The setup was placed on a desk with a height of 74 cm. The distance between user and display was 120 cm, the chair had a height of 50 cm. The uppermost visible line of the display was at 105 cm, and the screen was orthogonally aligned to the table.

To prevent users from looking at their fingers or the touchscreen we build a 45 $\times$ 32 $\times$ 23 cm^3 box with cutouts for the hands. Users were asked to put their hands with the device into this box, figure 5.2 shows the whole setup.

5.2.3 Participants

We recruited 24 users (aged 21–33, M = 24.95) for our study. 8 were female, 5 left handed, all reported regular smartphone usage. 72% of our users mentioned to use their phone in two handed landscape mode frequently.

5.2.4 Task & Procedure

We asked users to perform multiple sessions of a absolute indirect tapping task. Since we identified portrait mode to be not that useful with *HaptiCase*, we asked users to hold

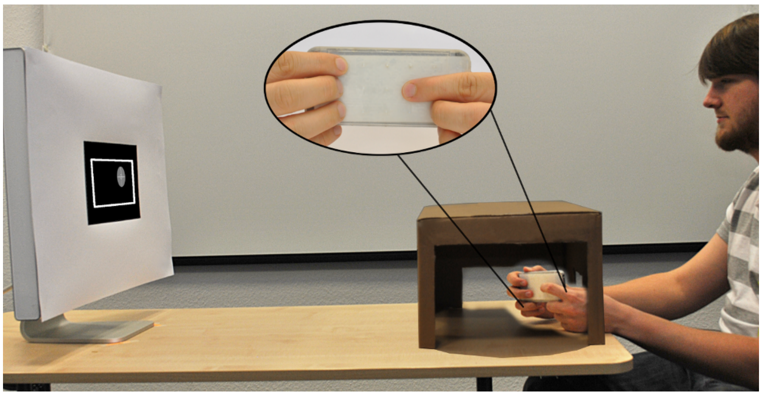

Figure 5.2: The study setup used during our first experiment. User held the device in landscape mode with index, middle, and ring finger on the back.

the device in horizontal landscape mode during the experiment. To touch the targets users used the thumbs of both hands, the other fingers could be used to utilize the tactile landmarks on the back of the device.

For each individual step a trial began when a target appeared on the distant screen. The targets were colored gray with a white cross hair, the screen background was black. Than users had to hit the corresponding area on the touchscreen of the input device. When users touched the screen they received a visual feedback where they hit the screen. The feedback was located at the touch position colored green, if the user hit, and colored red if they missed the target. A touch was considered to be a hit, when the touch was within the target size of 15×15 mm^2. When users lifted their finger the feedback faded out. To prevent accidental double hits, the system did not accept new touches until the fade out animation was finished.

The 15 mm target size on the device corresponded to 34.4 mm at the distant screen. When a target was hit, the user got presented the next target, if missed she had up to 4 additional attempts to retry hitting the target (as in Gilliot et al. [2014]). This way we kept a natural interaction, as users likely try to correct themselves after missing a target.

The log file was created on the computer to keep the interaction with the phone as responsive as possible. We logged the exact position of each touch and how close it was to the actual target. Also we logged if the user got feedbacked a hit or a miss, and the angles in which the phone was held, since we thought it might be a source for a systematic error users might make. Additionally we would be able to identify if any users hold the device in an undesired posture, like upside down or to much in portrait.

To estimate where their finger is placed, we told users to utilize the tactile landmarks on the back of the phone. Of course this was not possible with the *Base* design, since there were no landmarks on this design.

The distance between the outermost landmarks and the frame was not equal to the distance between two landmarks, since we distributed the landmarks equally from the center (Fig. 4.1). We added a visual outline of the outermost landmarks to the screen, since user probably assume all landmarks to be equidistantly distributed. Our idea was to offer users guidance not only from the frame, but also from the outermost landmarks.

Users were asked to perform the task as accurate as possible. When they felt, more preparation would help to hit more accurate they should feel free to take their time. After each set of trials, we asked users to fill out a short questionnaire and encouraged them to take a break before performing the next tasks.

5.2.5 Study Design

Independent variables for our first study were CASE, and TARGET. CASE was evaluated with six different levels, *Base*, *Frame*, *Dots Low*, *Dots High*, *Lines Low*, and *Lines High*. For TARGET we choose 28 different positions distributed over the screen. The TARGETs can be categorized in different ways. See figure 5.3 for the target collection, targets 0–17 were at least partially on the visual outline, 18–27 were within. Corresponding to the tactile landmarks, the tar-

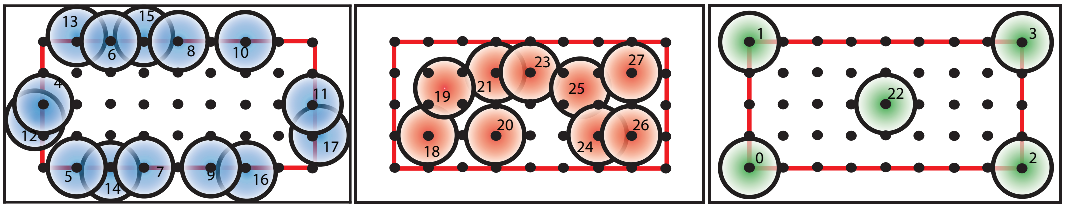

Figure 5.3: The collection of TARGETs with our categorization: left the *Border* targets, center the *Middle* targets, and right the *Key* targets.

gets were located either exactly on a dot or crossing for *Dots/Lines Low* (target 0–4, 6, 8, 10, 11, and 22), or on landmarks from *Dots/Lines High* (targets 5, 7, 9, 18, 20, 21, 23, 24, 26, 27). Targets 12–17 are located directly at the border of the screen, and target 19 and 25 were not located on a tactile landmark at all.

The categorization we used in our evaluation is shown in figure 5.3, we named three TARGET GROUPs: *Key* the group including all targets at the corners and the exact center point (0–3, 22), *Border* the group containing all targets that were close to the frame (4–17), and probably hardest to hit *Middle* targets (18–27, without 22). We assumed *Middle* to be the hardest group, as this would match the results by Gilliot et al. [2014].

We used a within group design where all users tested all CASEs and TARGETs. To counterbalance possible learning effects we used a balanced latin square design for CASE. The TARGETs were presented in pseudo random order.

Before each session, users were allowed to look at the upcoming CASE, we encouraged them to look at and feel the design before tapping eyes free with it. We also let users first tap a set of 10 training targets, before starting with the actual measurement. Each target was presented exactly once in each session, since the total amount of targets is 28, we had a sufficient amount of trials for each TARGET GROUP.

In total that resulted in at max 6 × 5 × 28 = 840 touches for each user, and in over 4000 datasets for our evaluation.

Depended variables for this study was OFFSET, the exact amount in mm from target center to the touch point center. We also evaluated ACCURACY which corresponds to the feedback users got, if they hit a target, and if not how many retries they needed.

Questionnaire

After each session, and at the end of the study we asked users to fill out a questionnaire. This questionnaire was designed to get some information, how users feel about *HaptiCase*, their impression of improving or worsening, and the strategies used with *HaptiCase*.

After each session we asked users on a 5 point likert scale, whether they exploited the tactile landmarks to orientate themselves, whether they found tactile landmarks made them more accurate in performing the task, whether *HaptiCase* made them more quickly, if the tactile landmarks confused more than helping, and if they think the amount was, too little, just right, or way too many. Additionally we used free text answering boxes, where users were asked to describe the strategy they used, what in particular was helpful, and what could be improved.

When users complete the whole set of trials we gave another questionnaire for the general impressions about *HaptiCase*. This questionnaire contained seven 5 point likert questions:

- Overall, I exploited haptic cues on the back of the device to find the targets on the touch screen.
- I felt comfortable using haptic cues on the back of the device.
- I prefer using haptic cues on the back of the device for blind touch interaction over not having any haptic cues.
- Overall, I had the impression that using haptic cues on the back of the device requires me more time to

find the targets.

- Overall, haptic cues on the back of the device helped me to find targets on the touch screen more accurately compared to having no haptic cues.
- Overall, haptic cues on the back of the device confused me more than helping to find the targets on the touch screen.
- I think I would perform more accurately by just looking at the device although nothing is displayed rather than navigating blindly with haptic cues.

All questions were ranked from totally disagree, to totally agree.

We also asked users to rank the six different designs on a 9 point likert scale, which design they liked most. At the end we offered an additional free text entry box for any further comments.

5.2.6 Results

For our evaluation, we only considered users first attempts to touch a target, regardless if the target was hit or not. We evaluated users SUCCESS RATE for the first hit on the 15 mm square targets, as well as the radial OFFSET from the targets center. We performed a detailed analysis on the different TARGET GROUPs. The idea was to find areas on the screen, where tactile landmarks are more or less useful than elsewhere. Additionally we evaluated users answers to the questionnaires, which strategy they used, and how comfortable they felt using *HaptiCase*.

Offset Values

For our OFFSET evaluation we only included users very first attempts on each target, no matter if this hit the target or not. To not disturb our measures by outliers, we removed touches that were away more than 3 $\times$ standard deviation from the respective mean. That resulted in 44 removals, corresponding to 1.1% of the data.

We report OFFSET as the euclidean distance (in mm) between a users touch, and the center of the presented target. The values for OFFSET by CASE were not normally distributed, and also not logarithmically-normally distributed, therefore we applied a aligned rank transform (ART) as suggested by Wobbrock et al. [2011]. We then used two-way repeated-measures ANOVA on the transformed data.

Target Offset by Case. We found a significant main effect of CASE on OFFSET ($F_{5,3797}$ = 8.06, $p < .0001$). Post-hoc analysis with pairwise comparisons showed that both *Dots* designs were significantly different from *Base*, *Frame* and *Lines High*.

There were no significant differences between the other designs. As expected *Base* performed worst (M = 7.63 mm), this corresponds to an average missing of the 15 mm target. *Lines High* and *Frame* performed slightly better (7.48 mm and 7.46 mm), with these design the users mean hit the target. Still the *Lines High* design performed second worst, what could indicate, that most users were confused by the amount of haptic cues in this design.

For *Dots Low* and *Dots High* the results were best, 6.75 mm for *Dots Low*, and 6.70 mm for *Dots High*. These to designs also had the best performance in success rate measures, both lead to more than 12% increase in first hit accuracy, compared to *Base*. Since both of these design let the users mean offset be about 1 mm shorter than the allowed offset for a target hit, 15 mm seem to be a reasonable target size when using these designs. Figure 5.4 shows a comparison for OFFSET ellipses for some targets. The red ellipses correspond to touch points when using the *Base* design, the green ones correspond to *Dots Low* trials. Interestingly, for both designs the Offsets were distributed into similar directions. The green ellipses for most targets are smaller than the red ones, since users were more accurate with the *Dot* designs.

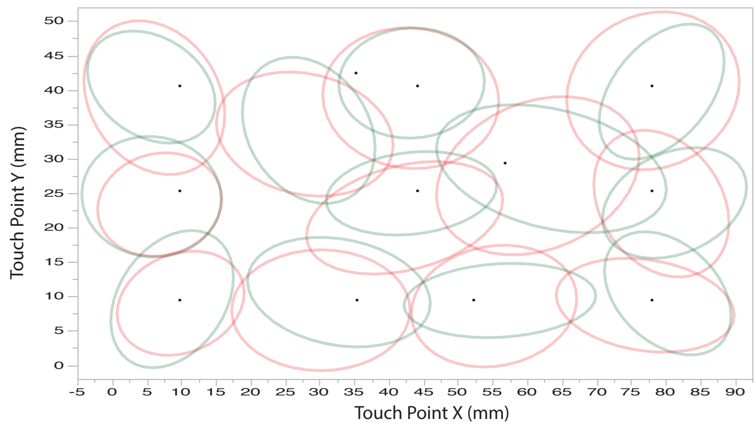

Figure 5.4: Ellipses for a 95% coverage of touches for a collection of targets. Touches generated with the *Base* design are colored red, touches from *Dots Low* are within the green ellipses. Original center points for the targets are black dots.

Both *Lines* designs, and the *Frame* design, did not perform significantly different from the *Base*, therefore we do not include these designs for more detailed analysis regarding the TARGET GROUP. Still *Lines Low* could be a interesting object for future analysis, since the boxes, that are created from the lines, were often perceived as buttons. Perhaps this design could lead to even better results, if the interface is more suited to a design like this.

	Offset			
CASE	Mean	SD	lo. 95% CI	up. 95% CI
Base	7.63	4.06	7.32	7.94
Frame	7.46	3.81	7.17	7.75
Dots Low	6.75	3.88	6.46	7.04
Dots High	6.70	3.60	6.43	6.98
Lines Low	7.16	3.91	6.86	7.46
Lines High	7.48	4.07	7.17	7.79

Table 5.1: Summary of the results in study 1 for OFFSET by CASE.

In Table 5.1 we list all mean, standard deviation and 95% confidence intervals, for OFFSET by CASE.

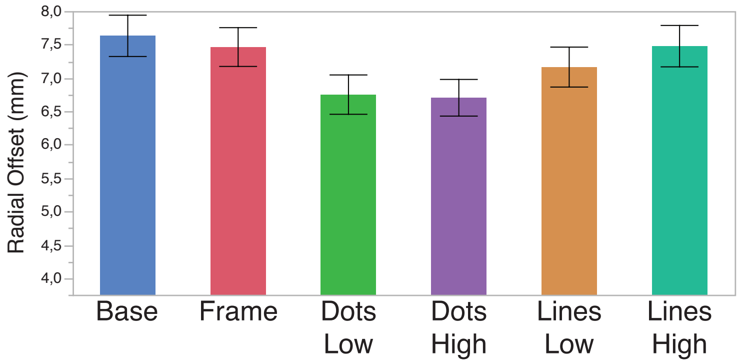

Figure 5.5: Mean OFFSET by CASE with 95% confidence intervals displayed.

Figure 5.5 shows a bar chart with mean OFFSETs for each CASE.

Target Offset by Target Group. To evaluate the effect of the target position on performance we categorized our TARGETs in three groups. See Figure 5.3 for the groups we created.

For the evaluation of OFFSET by TARGET GROUP, we focused on the designs that had significant differences in the previous evaluation. These designs were *Base* with the lowest result, and *Dots Low / High* with the best results.

e compared *Base* against *Dots Low / High* regarding the users' target offset for our TARGET GROUPs. Fig. 5.6 illustrates mean OFFSETs by TARGET GROUP. *Dots High* performed significantly better than *Base* in all the conditions ($p < 0.05$, each). Though for *Dots Low* this was only true for *Key* and *Border* targets ($p < 0.01$, both). For *Middle* targets p was 0.0631, therefore we still can report a trend to better performance with *Dots Low*.

Comparing the different TARGET GROUPs, *Key* targets were hit the best(< 5.61 mm). As we expected, and confirming Gilliot et al. [2014], *Border* targets were hit more precise than *Middle* targets (< 6.49 mm for *Border*, and < 7.99 mm for *Middle*). Hence, the target size of 15 mm was not sufficient for the average OFFSET to hit the target.

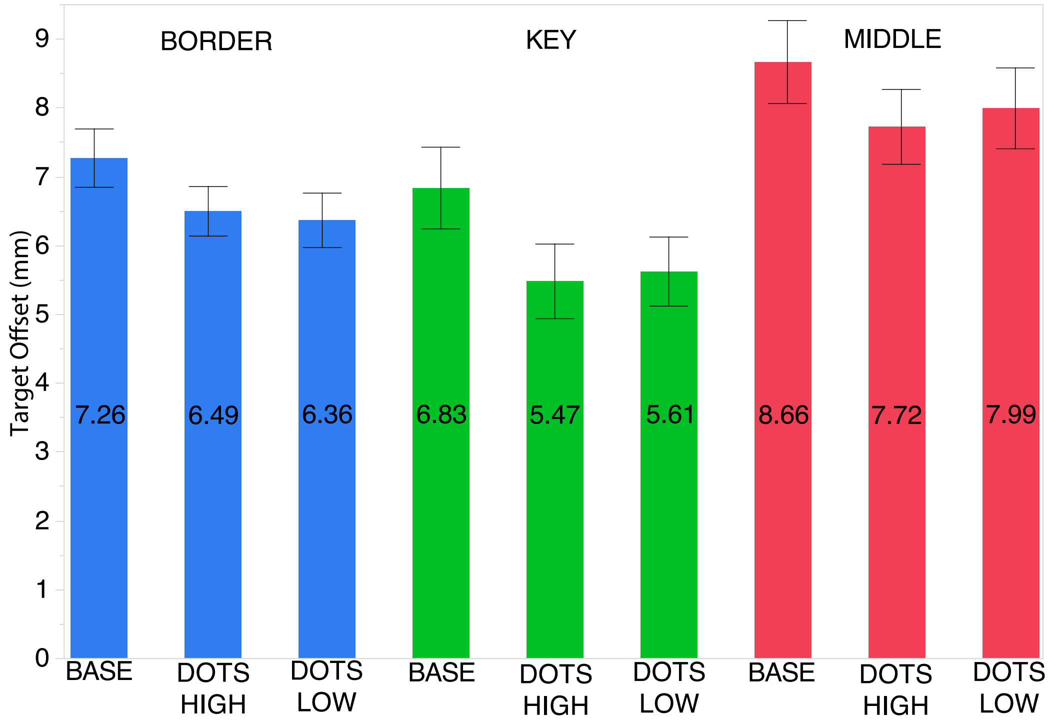

Figure 5.6: Text

Interestingly, a closer look to the TARGET GROUPs revealed, that targets from the *Border* group, that were located at the top of the screen (targets 6, 8, 10, 13, and 15), were significantly harder to hit (OFFSET 7.08–7.87 mm). We called the group of targets *Top*. Table 5.2 shows the comparisons between *Top* and *Border* results from our participants.

CASE	Offset *Top* [mm]	Offset *Border* [mm]	p
Base	8.54	6.56	$< .0001$
Dots Low	7.12	5.94	$= .0199$
Dots High	7.19	6.11	$= .0033$

Table 5.2: Mean Offsets, and p values, for *Top* and *Border* targets

A possible explanation for this effect could be the fact, that *Top* targets are farthest away for the users thumbs to reach. When also needing to hold the index finger on the back, the pinch gesture is not that accurate anymore for targets far away. Though users faced these difficulties for these targets, they still performed significantly better when using *Dots High* or *Dots Low* than with the *Base* design (M = 7.19, and 7.12 mm against 8.54 mm $p < .001$).

User Feedback

User strategies. We asked our users to explain the strategy they used to hit the on screen targets. We wanted to confirm, if users felt they were able to use *HaptiCase* as described in chapter 2.

For Base, 15 users reported to have used pure guessing to reach the targets. Some users mentioned that they tried to set their fingers as constraints to orientate themselves in future trials. For *Dots Low*, 15 users moved their fingers over the tactile landmarks to find the correct position on the screen, and afterwards used a proprioceptive pinch to hit the target. By contrast, six users fixed their index, middle-, and ring finger on the tactile landmarks and hit targets by approaching these fingers with the thumb instead of moving a finger around on the back of the device. Two users mentioned that they did not use the landmarks frequently, since they felt uncomfortable using the smartphone in landscape mode.

User Confusion. We asked users to tell us if they felt confused by the tactile landmarks. Overall users mentioned *HaptiCase* to be only little confusing (M = 2.00 SD = 0.33 on a 5 point likert scale). When taking a look into the rating for each design separately, we found that the *Lines High* design confused users more than helping (M = 3.52 SD = 1.30 on a 5 point likert scale). Table 5.3 names mean and standard deviations from users answers on whether they were confused or not.

The *Dots Low* design confused users least, as this also is best performing, we decided to use this design for our second user study.

CASE	Mean	SD
Overall	2.00	0.33
Frame	2.00	1.17
Dots Low	1.60	0.98
Dots High	1.96	1.11
Lines Low	1.91	0.95
Lines High	3.52	1.31

Table 5.3: Mean and standard deviations for the 5 point likert question, whether *HaptiCase* confused the users

5.2.7 Summary

In our first study, we can accept hypothesis 1 and 3. When using *HaptiCase*, users are more accurate in an eyes free absolute indirect tapping task. In detail, we found the *Dots Low* and *Dots High* designs to be most accurate. *Lines Low* had good results as well, but as shown in the *Lines High* design, continuous landmarks can be confusing as well. The center target and targets and the corner were most accurately hit, followed by targets at the border, and targets towards the middle. We had to reject hypothesis 2, higher resolution of landmarks did not increase users accuracy. Therefore, we concentrated on the DotsL design for our second study.

5.3 Experiment 2 on *HaptiCase* With and Without Vision

To investigate how *HaptiCase* influences user performance and experience in tapping task, when it is possible to look at hands and the device, we performed a second user study. Our idea was that most users will be more accurate when they are able to see the device, even when nothing is displayed on the touchscreen. On the other hand we also thought, that *HaptiCase* will not significantly worsen users performance when looking and using tactile landmarks.

In addition we wanted to get insight in users timing performance when they use *HaptiCase*. We measured timings for pure visual interaction, pure tactile guided interaction, and the combination of visual and tactile. Here we though that *HaptiCase* will probably make the users less fast, since the tactile tracking state needs some time to be performed.

5.3.1 Hypotheses

For our second study, we hypothesized that the combination of visual and tactile targeting will not significantly differ from pure visual targeting. For the timing we hypothesized, that users will need more time to perform the task when tactile targeting is available.

5.3.2 Apparatus

For our second experiment we used nearly the same setup as in 5.2.2. Again our users used the 4" screen (1136 $\times$ 640 px^2) iPhone 5S, and sat at the same desk setup. For the conditions where users were allowed to look at the screen, we removed the visual barrier for the hands. The screen was left at blank white, this way users could easily see where the screen is located, but no actual targets were displayed on the touchscreen. We used the same screen and bezel as in our first experiment.

Additionally, we removed the visual outline from experiment 1. We wanted to compare the eyes-free results from study 1 and study 2, to see whether this outline had an influence on users accuracy in study 1 or not.

Again all measurements and the data logging was taken on the computer. The timing measurement was taken on the computer as well, using the same clock for all time stamps. This way the connection delay adds to the response time of the users. We choose to measure time only on one device, since pairing of the devices was not possible, and the magnitude of the delay is way smaller than the response time of the users.

Participants

For our second study we recruited twelve additional users. These were aged from 20 to 36 (M = 24.74), one was left-handed, and five were female. One of these reported to not own a smartphone, but work with them on regular base. To preclude learning effects between the two user studies, none of these participants participated in our first study.

Questionnaire

For the second study users filled out a questionnaire after they performed all tapping tasks. In this questionnaire we asked users to rank with which design they performed, most accurate, most quickly, and which worked best for their feeling. We also offered a free form box to give any other insight, how often they looked at the phone or which strategy they used.

In addition we observed how often users looked down into the direction of the phone, this observation however, was only informal, to see weather any participants do not look at all.

5.3.3 Study Design

In this study our factors were CONDITION, and TARGET. CONDITION was evaluated with three different levels: The *Base* design with vision (*BaseVis*), the *Dots Low* design with and secondly without vision (*DotsVis*, and *DotsNoVis*). The last one *DotsNoVis* was used to compare the results with the first experiment. Since the second study focused more on the influence of the vision, we only used the *Dots Low* design, which performed really good, and also was evaluated best by most users.

We counterbalanced CONDITION with a balanced latin square design, TARGETs were presented in pseudo random order. We used the same target set as in our first study (see Fig. 5.3).

For our second study the dependent variables were: OFFSET, as in study 1, and TIME, which corresponds to the time users needed to respond on a presented target on the screen. This measure was taken on the computer only, to circumvent possible problems with the device pairing. We assumed the connection delay to be to small and uniform to influence our results in a undesired way.

5.3.4 Results

Condition and Target on Target Offset As in study 1, we analyzed the effect of CONDITION and TARGET on OFFSET. Since OFFSET was neither normally distributed, nor logarithmically-normally distributed, we applied an Aligned Rank Transform Wobbrock et al. [2011]. A repeated measures ANOVA showed that TARGET did not have a significant main effect on OFFSET and there was also no TARGET × OFFSET interaction effect. However, we found a significant main effect of CONDITION on OFFSET ($F_{2,913}$ = 26.50, $p < .0001$). Post-hoc comparison using a Tukey HSD showed significant differences for OFFSET between *Dots Low NoVis* and *Base Vis* and between *Dots Low NoVis* and *Dots Low Vis* (both $p < .0001$). Table 5.4 shows mean and standard deviations for OFFSET per CONDITION.

CONDITION	Mean[mm]	SD[mm]	lower 95% CI[mm]	upper 95%CI[mm]
Base Vis	5.36	3.42	4.99	5.72
Dots Low NoVis	6.99	3.62	6.60	7.38
Dots Low Vis	5.66	3.64	5.27	6.05

Table 5.4: Summary of the results for OFFSET in our second experiment

When being able to look at the device, users were on average 1.63 mm (for *Base Vis*) more accurate compared to condition *Base Vis*. For *Dots Low Vis* users were 1.33 mm more precise. There was no significant difference between the two vision conditions.

Comparing the results from *Dots Low* in our fist and second users study, we found the targeting offset to be increased by 0.24 mm in experiment 2 (6.75 mm in study 1 compared to 6.99 mm in study 2). However this results still fits within the upper bound of the 95% confidence interval from study 1. A one-way independent measures ANOVA showed no significant differences between the two studies for this condition ($F_{1,1004}$ = 1.63, p = 0.2018). This confirms, that the visual outline on the screen

Furthermore, a one-way independent measured ANOVA comparing target offset (aligned rank-transformed) for this condition between Studies 1 and 2 was not significant ($F_{1,1004}$ = 1.63, p = 0.2018), confirming that the target accuracy was almost the same for both studies for *DotsL*. Since we omitted the visual outline in this study, but the results were similar to study 1, we can conclude, that the visual outline in study 1 had no significant influence on users accuracy.

Condition and Target on Time In this experiment, we also measured the time the user needed to perform the touch after the target was shown on the display. TIME on CONDITION was neither normally distributed, nor logarithmically-normally distributed. Thus, we applied the Aligned Rank Transform for TIME Wobbrock et al. [2011]. Similar to the OFFSET results, we included timing data from all first attempts, regardless if the target was hit or missed. We conducted a repeated measures ANOVA on the transformed data. Interestingly, there was no significant main effect of TARGET on TIME, and there was no CONDITION × TARGET interaction effect. However, there was a significant main effect of CONDITION on TIME ($F_{2,994}$ = 20.98, p $<$ .0001). Tukey HSD post-hoc comparison showed a significant difference between *Base Vis* and *DotsL NoVis* and between *DotsL Vis* and *DotsL NoVis* (both $p < .0001$).

Table 5.5 shows mean and standard deviation for TIME per CONDITION. As we hypothesized before, users were slowest, when using only *HaptiCase* without being able to see the device (+334.88 ms compared to *Base Vis*). When users were able to utilize both, the visual and the tactile sense, they were 126.58 ms slower compared to not having tactile

CONDITION	Mean[ms]	SD[ms]
Base Vis	1510.30	386.11
Dots Low NoVis	1845.18	881.74
Dots Low Vis	1636.88	636.44

Table 5.5: Summary of the results for TIME in our second experiment

landmarks. However, the difference was not significant for *Base Vis* and *Dots Low Vis*.

User Feedback

As we expected, users preferred looking at the device for accurate interaction. On a 5-point Likert scale (increasing from 1 to 5) *Base Vis* scored 4.00 (*SD*=0.85) on average as did *Dots Low Vis* (*SD*=1.35). *Dots Low NoVis* scored only *M*=3.17 (*SD*=1.11).

However, a Friedman analysis (χ^2=7.61, *df*=2, *p*= 0.022) showed, these results to be not significant. Four users did rarely look at the input screen when performing the task, but the others looked frequently at the device to place their finger.

5.3.5 Summary

Study 2 confirmed both our hypotheses: Users are more accurate when being able to look at the input device, and are able to hit the targets faster.

HaptiCase cannot replace the visual sense in terms of accuracy in an absolute indirect pointing task. However, when users are not able to look at the input screen, tactile landmarks significantly increase users accuracy in these tasks.

Chapter 6

Implications for Designers

Since we showed that *HaptiCase* improves user's accuracy in eyes-free tapping tasks, we suggest using tactile landmarks on the back of devices to enable eyes-free interaction. In this chapter, we give implications for designers, who choose to use *HaptiCase* for their project. On the one hand, this includes designers who want to create tactile landmarks to enable eyes-free use of smartphones. On the other hand, application designers can use our insight as well. An application designed to be controlled with tactile landmarks, could be used eyes-free in everyday interaction.

HaptiCase can be utilized by phone case designers and application designers.

We will name three different approaches to design a *HaptiCase* or applications with tactile landmarks. The first to be general implications on tactile landmark designs, that can be drawn from our experiments and the users feedback. Implications on how a *HaptiCase* could be designed when it should serve for a specific application will be named as well. Thirdly, we will name design suggestions regarding how an application that is meant to work eyes-free could be designed. The graphical user interface of such an application would be adjusted to spacial constraints of tactile landmarks.

We name three approaches to design with *HaptiCase*.

6.1 Universally applicable *HaptiCase* designs

There are different potential scenarios, in which *HaptiCase* could offer valuable support. All these scenarios share the users wish, to interact with the device eyes-free. Designers, who want to create a *HaptiCase* that should be useful in many different situations should have the possible scenarios in mind as well.

The Frame is a very important landmark.

In general, our users mentioned the frame corresponding to the touchscreen as really useful. Therefore, it should always be included in a *HaptiCase* design. Leaving out this landmark, should be considered carefully. Because of the frame, users were able to tell if their thumb will hit the screen or not, just by holding the device in their hands.

Many users gave interesting design recommendations.

Other than that, our experiments as well as the user feedback, showed that concrete landmarks are more useful than continuous cues. Some users suggested to exchange our dots with small circles, but this still more corresponds to the *Dots* designs, than continuous lines. A comparison of low and high resolution designs accuracy showed, that adding more than a 3×5 grid of tactile landmarks does not improve the accuracy. Therefore, we recommend to keep the amount of tactile landmarks within this range, to not confuse the users.

For everyday usage, *HaptiCase* should be designed more generic.

We further recommend to keep a generic design, since it should fit for a variety of interfaces. As shown in chapter 5.2.6, users can accurately hit the corners and the center of the screen. Landmarks at these positions can be used as starting reference points. From there on, other landmarks can be used for more detailed selection. Some users mentioned they would like to have a landmark design with static reference points for their index and middle fingers. Figure 6.1 illustrates such a design, as recommended by some users. The index fingers would be placed steadily in the red landmarks, while the middle fingers remain in the blue landmarks. The strategy to reach a target with this design is a little different to our initial idea. A user would place her index and middle fingers within the circu-

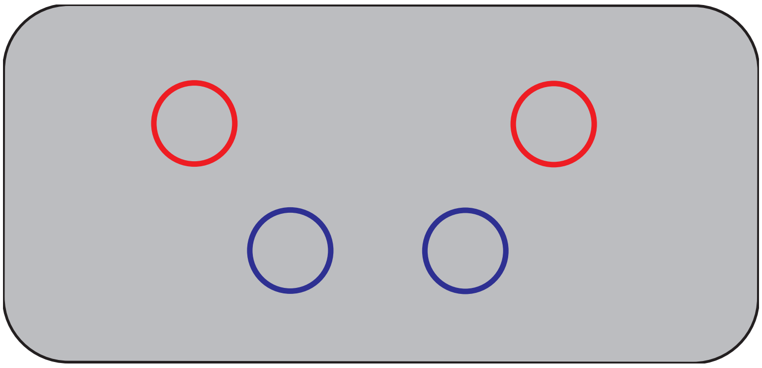

Figure 6.1: A *HaptiCase* design as suggested by several users the circular landmarks are meant to be resting points for index (red) and middle fingers(blue).

lar landmarks, and keep their fingers at this position. Subsequently, each on-screen target is reached by positioning the thumb relative to these concrete positions.

6.2 Designing *HaptiCase* for specific Applications

When designing a *HaptiCase* for a specific application, we do not need to be as generic as possible. A first step would be to identify frequently accessed functionality of the application, especially functionality that is meant to be accessed eyes-free.

After identifying the areas on the screen that are often touched, we recommend to add tactile landmarks at the corresponding positions on the back. With a tactile landmark, users will be able to find the position of the on-screen targets more accurate. Our recommendation would be to do this for most outstanding functionality of the application. An application with such a tailored *HaptiCase* will most likely be accessed more accurate in eyes-free interaction.

Application specific *HaptiCase*s can be less generic.

During our experiments, some users mentioned they

Figure 6.2: A graphical user interface, as it could be used to steer a plane in a gaming application.

It is important to identify most frequently used functionality.

would like to match the interface as tactile landmarks. However, this would include tactile landmarks for every possible on-screen target. We would recommend to not simply mirror the graphical user interface of an application to tactile landmarks, since this might be more confusing than helpful. Figure 6.2 shows an example for a gaming application[1] on a tablet computer. Mirroring this interface to a tactile landmark on the back, will possibly overwhelm users. Instead, frequently accessed targets should be made haptically explorable.

For quick accurate access, designers should use as few landmarks as possible.

Our recommendations on designing a *HaptiCase* also hold for eyes-free interaction with memorized applications. While driving a car, users cannot switch their attention to their smartphone, since they need to concentrate on the traffic. In this situation, incoming phone calls have to be rejected. There are applications that allow users to not only decline the call, but also send pre-defined text messages as answer to the caller. A *HaptiCase* for such an application could offer the starting point for the series of inputs, to decline the call and send the message. There would be no

[1] http://z2.com/game/metalstorm-online/

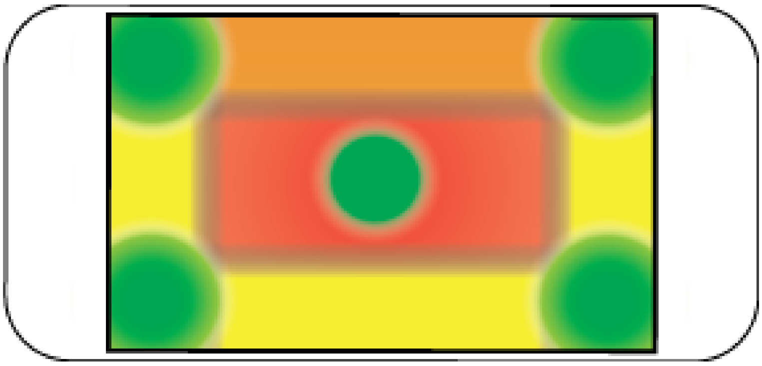

Figure 6.3: Regions we identified to be interesting in application design with *HaptiCase*(Green areas are hit most accurate, red areas least accurate).

other landmarks, since the user should be able to quickly grasp the phone and perform the set of gestures, not being distracted by unnecessary landmarks.

6.3 Application design with tactile landmarks

When designing an application that is meant to be accessed eyes-free, application designers can benefit from our results as well. Based on the results from our experiments, we identified certain areas on the screen to be easily accessible with tactile landmarks. These areas could be used for functionality that is accessed frequently. Since the users accuracy is higher in these areas, application designers can reduce the target size there. Towards the middle of the screen users are less accurate, therefore designers should increase target sizes in this area.

6.3.1 Region suggestions

Figure 6.3 shows a map of touch regions on the screen. Green colored areas were hit most accurately, whereas red areas were hit least accurately. As evaluated in chap-

We identified interesting region features.

ter 5.2.6, the corners of the screen and the center point were hit most accurately. These areas are good candidates for functionality that is accessed frequently. Due to our results, a mean target size of 12.2 mm would be sufficient for targets to be hit reliably on the first attempt.

Functionality that is accessed less frequently, could be placed in the yellow areas. Users are still able to hit targets at a size of 12.9 mm reliably in this area, therefore this are can be used for important but less frequently accessed targets. Though this is true for targets at the border, the top border is an exception for this (Orange area in Fig. 6.3). Targets within this area are hit less accurately by users, compared to other border targets. A possible explanation for this effect is, that the top border is the farthest to reach with the thumbs. Targets in this area need a minimal size of 14.9 mm to be hit accurately.

Targets at the top and the middle are harder to hit than border and key targets.

The most difficult region to hit, is around the center point(Red area in Fig. 6.3). Designers should carefully use this area for on-screen targets. The minimal target size users need in this area is 15.8 mm.

6.3.2 An Example Application Designed with Tactile Landmarks

To demonstrate how our design recommendations could be applied, we created a graphical user interface for a TV remote control application (Fig. 6.4). This application is not meant to replace a full remote control, but instead it offers quick eyes-free access to frequently used functionality.

The volume control, channel switch, and time shift buttons are placed in areas that are easy to reach. We identified these functions to be most useful when placed in the area for frequently used targets. Along the side borders we placed a bigger mute and return to the last channel button (PP in Fig. 6.4). Buttons can still be accessed accurately since they are bigger than the previously mentioned buttons. At the bottom border we placed the colored function buttons, as found on typical remote controls, these are

A TV remote application could look like this.

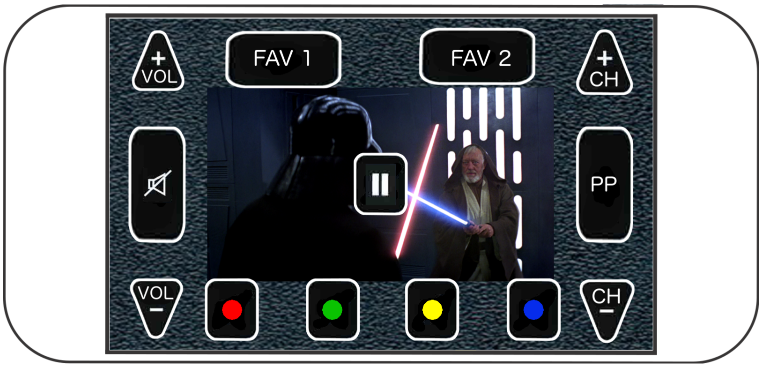

Figure 6.4: An example for a touch-based TV remote application, designed following our recommendations for on-screen targets.

needed for example in interactive Television applications. Since the top border is harder to hit, we used it to position two buttons where favorite channels can be stored for shortcut access. Towards the center, we did not add any other on-screen targets, since this area is hard to be hit accurately.

Figure 6.5 shows a possible *HaptiCase* design for this application As in the designs from Chapter 4.1 a frame indicates the borders of the touchscreen. Five small dots indicate the key positions, corresponding to the most frequently used buttons. The other areas are separated with thin lines, to enable users differentiating whether they are at an on-screen button or not.

We did not add additional landmarks for the targets at the side borders or the bottom, since we wanted to keep the design simple as well. Users should be able to identify each on-screen button, since it is either placed directly at an tactile landmark, or adjacent to it. This design could serve in everyday interaction with Apples iOS as well, as the borders are often used as navigation bars. Users could use the circular landmarks at the corners, to find ”Next” and ”Previous” buttons in Apples navigation bars.

In our example application, we implemented most of our design recommendations. However, that does not mean

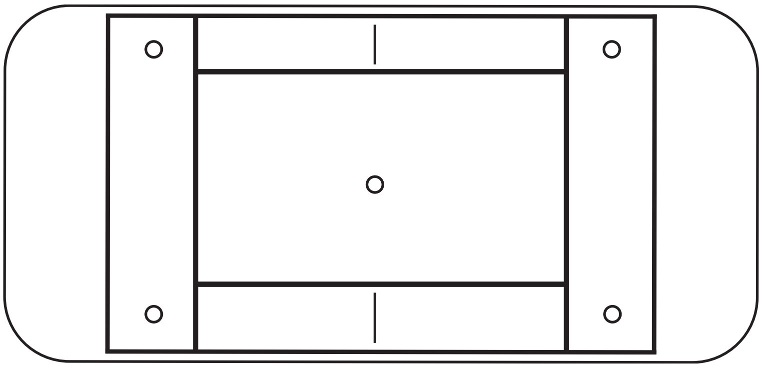

Figure 6.5: A *HaptiCase* design, that could be used with the TV remote application.

every application has to look like this, to work with tactile landmarks. Our region suggestions are applicable for designers who want to plan where frequently used buttons should be placed. An application that requires fewer onscreen buttons, might not need to fulfill our rules for easy access. We focused mainly on tapping interaction. Designers who create an interface based on swipe gesture interaction, might use a completely different *HaptiCase* design.

Chapter 7

Summary and Future Work

In this thesis we presented a novel interaction technique that uses tactile landmarks to offer users feed-forward guidance for eyes-free tapping tasks. To conclude the thesis we will name ideas, how future work could strengthen our results, or investigate interaction scenarios with *HaptiCase*.

7.1 Limitations

Lab experiments likely are not able to represent everyday situations.

Though designed and planned with care, lab studies suffers some limitations. We performed our experiments with 36 different people, though this number is sufficient to get meaningful results, the conclusions we drew could be stronger when there had been more users testing *HaptiCase*. Our eyes-free absolute indirect tapping task was reduced to a very simple interface. There was only one target present at each moment in time, and the screen was free from other disturbances. This situation will rarely happen in everyday interaction, and therefore limit our conclusions on everyday scenarios with *HaptiCase*. Our results are based on one specific smartphone as well. Though the iPhone 5S we used, is a common used smartphone, other devices, with bigger boundaries or thicker devices might lead to differ-

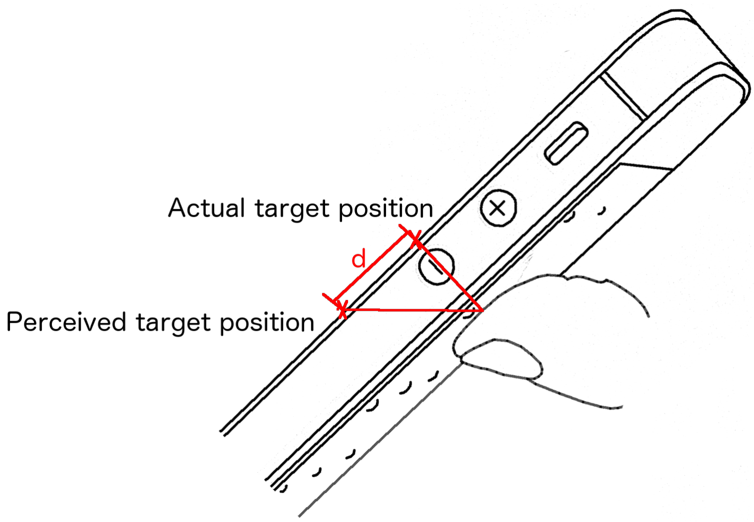

Figure 7.1: The Pinch Offset Effect that could explain for consistent errors between users.

ent results.

The Pinch Offset Effect could be an interesting research topic.

The Pinch Offset Effect Regarding the thickness of the smartphone we found another possible research area. Since our interaction is based on the proprioceptive pinch, with the device laying between the pinching fingers, the thickness of the device should be taken into consideration as well. When aiming for the finger on the back side, users hit the screen slightly before they expect it, due to this device thickness. Since the device is hold at a certain angle, there might be a consistent error between users targeting wit the tactile sense. Figure 7.1 shows a sketch for this effect.

We were aware of this possibility before performing the user studies, and therefore logged the angle in which users held the input device. However, in our experiments, we could not find any systematic error that correlated with the angle of the phone. Nevertheless this effect could be reviled, when performing a user study that aims to investigate this effect.

We did not investigate how *HaptiCase* influences users performance, when users are able to train the interaction technique more. Therefore, we could only make assumptions, on *HaptiCase* performance with memorized spacial layouts. Users might be a lot more accurate and fast, when training eyes-free interaction guided by the tactile sense. In our experiments we let users look closely on each design, and gave them some test trials before the measurement started, but this cannot substitute weeks or month of training with *HaptiCase*.

7.2 Future Work

We have several ideas, how researchers could build upon our work for future research. Investigating the potential of tactile landmarks when users train the interaction for a longer time would be interesting. When users are able to use *HaptiCase* for several weeks, for example using a *HaptiCase* supported remote control application, it might be accurate enough to compensate for missing eye-contact even more. Longer testing also might lead to interesting new design ideas for *HaptiCase*, during our studies some users already mentioned improvement ideas for the tactile landmarks. Taking our designs one step further, and trying to find a *HaptiCase* design, that truly is useful in everyday interaction would be another interesting idea. Do trained users prefer *HaptiCase* designs with less landmarks, or do additional landmarks increase in accuracy over time, might be questions that could be answered as well.

Another interesting approach could be to take a look, how *HaptiCase* can influence application design even further. We identified regions that are useful for landscape orientation of the phone. It might be possible, to create designs that work at portrait mode as well.

We designed our tactile landmarks with care, but still there are a lot of other possibilities to design a *HaptiCase*. During our user studies many users came up with own design ideas, some of them are truly worth to be considered for future *HaptiCase* designs.

7.3 Summary and Contribution

With this thesis we introduced a novel interaction technique for eyes-free *absolute indirect* tapping tasks with a mobile touchscreen as input device. Though this task is normally difficult to perform, since users are not able to orientate on a flat touchscreen without looking at it, we measured a significant performance increase when offering tactile landmarks on the back of the device.

We explained the interaction technique in detail, and integrated it into the well known graphical state chart models by Buxton [1990]. *HaptiCase* offers a tactile targeting state, in which users can explore the physical dimensions of the screen, before tapping an on screen target. Utilizing *HaptiCase* users are able to interact more accurate with touchscreens, when they are not able to see the input device.

We created eight different tactile landmark designs and explained how designers or researchers with access to a laser cutter could create their own *HaptiCase*. To evaluate our different design ideas, we performed a series of user studies and reported the results in chapter 5. Users were performing an eyes-free tapping task, with tactile landmarks on the back of the input device. Users accuracy increased significantly compared to the landmark-free baseline.

We named several implications for case designers, as well as application developers, which can be followed if someone wants to create a *HaptiCase*, or an application that should be used eyes-free. These implications addressed designers, who want to create a generic *HaptiCase* for a variety of tasks, as well as designers, who want to create a case for a specific application.

Appendix A

Appendix for the eyes-free tapping accuracy experiment

These are the questionnaires we handed out during our first experiment.

The target collection we used during our experiments

HaptiCase (before):

Participant ID:

1. Gender: ◯ Male ◯ Female
2. Age: ___________________
3. Which smartphone model(s) do you use? (Do you know the screen size?)

____

4. What is your dominant hand? ◯ Left ◯ Right
5. For each of the activities below, please indicate:
   A. Which hand you prefer for that activity?
   B. Do you ever use the other hand for the activity?

   Which hand do you prefer to use when:

	Left	Right	Do you ever use the other hand?
Writing	◯	◯	☐
Drawing	◯	◯	☐
Throwing	◯	◯	☐
Using Scissors	◯	◯	☐
Using a toothbrush	◯	◯	☐
Using a knife (without a fork)	◯	◯	☐
Using a spoon	◯	◯	☐
Using a broom (upper hand)	◯	◯	☐
Striking a match	◯	◯	☐
Opening a box (holding the lid)	◯	◯	☐
Holding a computer mouse	◯	◯	☐
Using a key to unlock a door	◯	◯	☐
Holding a hammer	◯	◯	☐
Holding a brush or comb	◯	◯	☐
Holding a cup while drinking	◯	◯	☐

Figure A.1: The questionnaire we handed out to our users before the experiment

HaptiCase (in between):

Participant ID: Backside Design: 0

1. If applicable, please describe your **strategy** used to find the targets on the touch screen. If you exploited haptic cues, please specify how you navigated through them.

7. What in particular did you find **helpful** for this design?

8. What could be **improved** for this design?

Figure A.2: The questionnaire we handed out to our users after the *Base* design

HaptiCase (in between):

Participant ID: Backside Design:

	totally disagree		neither		totally agree
1. I exploited the haptic cues frequently to find the targets on the touch screen.					
2. The haptic cues were helpful in finding the targets on the touch screen **quickly**.					
3. The haptic cues were helpful in finding the targets on the touch screen **accurately**.					
4. Overall, the haptic cues confused me in finding the targets on the touch screen.					

	way too little		just right		way too many
5. The amount of haptic cues was					

6. If applicable, please describe your **strategy** used to find the targets on the touch screen. If you exploited haptic cues, please specify how you navigated through them.

7. What in particular did you find **helpful** for this design?

8. Regarding this design: what could be **improved**?

Figure A.3: The questionnaire we handed out to our users after each landmark design

HaptiCase (afterwards):

Participant ID:

1. Overall, I exploited haptic cues on the back of the device to find the targets on the touch screen.

totally disagree		neither		totally agree

2. I felt comfortable using haptic cues on the back of the device.

totally disagree		neither		totally agree

3. I prefer using haptic cues on the back of the device for blind touch interaction over not having any haptic cues.

totally disagree		neither		totally agree

4. Overall, I had the impression that using haptic cues on the back of the device requires me **more time** to find the targets.

totally disagree		neither		totally agree

5. Overall, haptic cues on the back of the device helped me to find targets on the touch screen more **accurately** compared to having no haptic cues.

totally disagree		neither		totally agree

6. Overall, haptic cues on the back of the device confused me more than helping to find the targets on the touch screen.

totally disagree		neither		totally agree

Figure A.4: The questionnaire we handed out to our users after the experiment

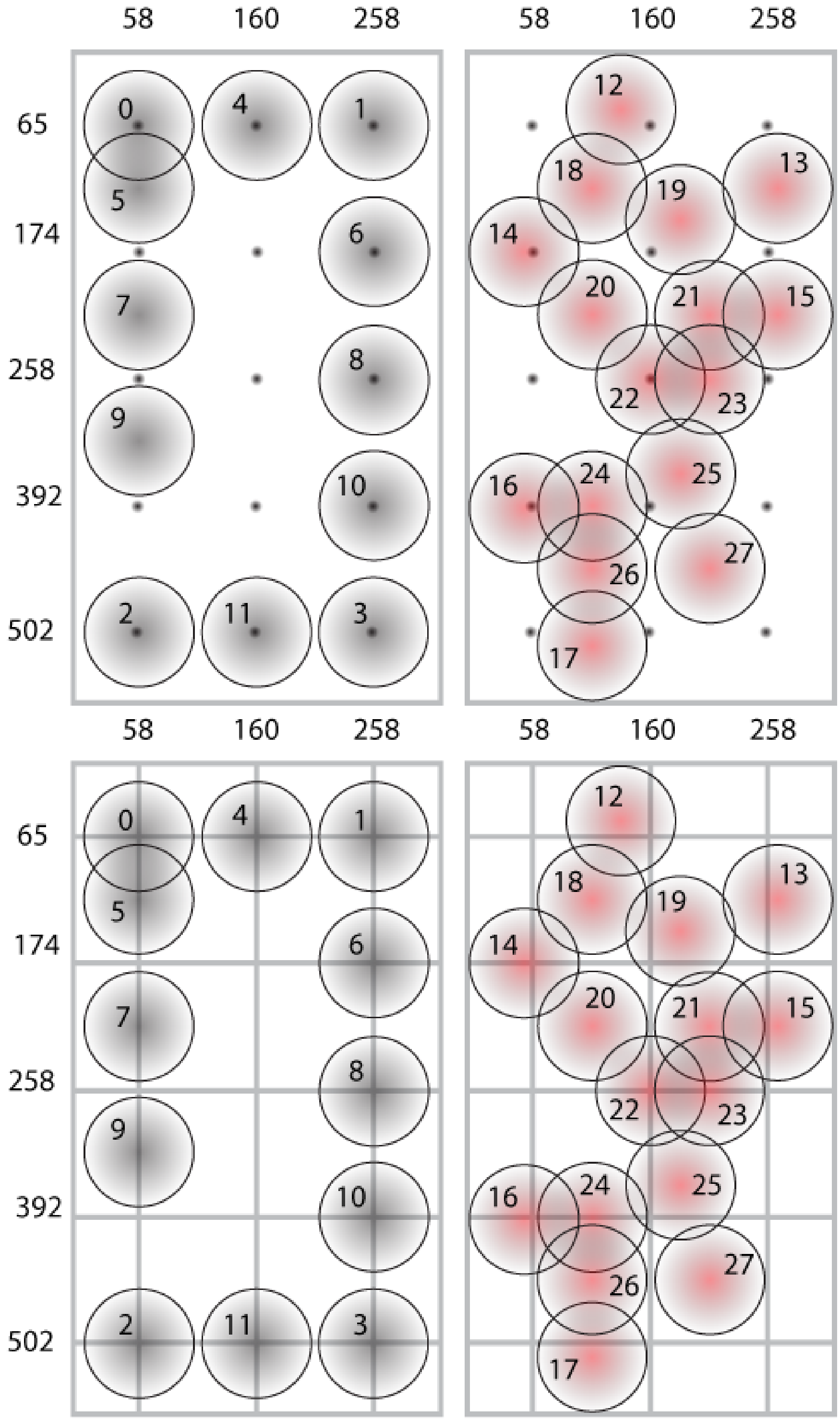

Figure A.5: The target collection for our experiments

Appendix B

Appendix for the second tapping accuracy experiment

The questionnaires from our first study were slightly changed for the second experiment.

HaptiCase 2 (before):
Participant ID:

1. Gender: ◯ Male ◯ Female

2. Age: ___________________

3. Which smartphone model(s) do you use? (Do you know the screen size?)

____

4. What is your dominant hand? ◯ Left ◯ Right

Figure B.1: The questionnaire we handed out to our users before second the experiment

HaptiCase 2 (afterwards):

Participant ID:

1. For each of the conditions: I was able to perform the task **accurately**.

Backside design	totally disagree		neither		totally agree
blank without haptic cues with vision					
dots without vision					
dots with vision					

2. For each of the conditions: I was able to find the targets **quickly**.

Backside design	totally disagree		neither		totally agree
blank without haptic cues with vision					
dots without vision					
dots with vision					

Figure B.2: The questionnaire we handed out to our users after the second experiments

Bibliography

Olivier Bau, Ivan Poupyrev, Ali Israr, and Chris Harrison. TeslaTouch: Electrovibration for Touch Surfaces. In *Proc. UIST*, pages 283–292, 2010.

Patrick Baudisch and Gerry Chu. Back-of-Device Interaction Allows Creating Very Small Touch Devices. In *Proc. CHI*, pages 1923–1932, 2009.

William Buxton. A Three-State Model of Graphical Input. In *INTERACT*, 1990.

Maria Claudia Buzzi, Marina Buzzi, Francesco Donini, Barbara Leporini, and Maria Teresa Paratore. Haptic Reference Cues to Support the Exploration of Touchscreen Mobile Devices by Blind Users. In *Proc. CHItaly*, pages 28:1–28:8, 2013.

Tom Carter, Sue Ann Seah, Benjamin Long, Bruce Drinkwater, and Sriram Subramanian. UltraHaptics: Multi-Point Mid-Air Haptic Feedback for Touch Surfaces. In *Proc. UIST*, pages 505–514, 2013.

Yasmine N El-Glaly, Francis Quek, Tonya Smith-Jackson, and Gurjot Dhillon. Touch-Screens are Not Tangible: Fusing Tangible Interaction with Touch Glass in Readers for the Blind. In *Proc. TEI*, pages 245–252, 2013.

Brian Frey, Caleb Southern, and Mario Romero. BrailleTouch: Mobile Texting for the Visually Impaired. *HCI*, 6767(Chapter 3):19–25, 2011.

J Gilliot, G Casiez, and N Roussel. *Impact of Form Factors and Input Conditions on Absolute Indirect-Touch Pointing Tasks*. 2014.

Cagatay Goncu and Kim Marriott. GraVVITAS: Generic Multi-touch Presentation of Accessible Graphics. In *Proc. INTERACT*, pages 30–48, 2011.

Tiago Guerreiro, Joaquim Jorge, and Daniel Gonçalves. Exploring the Non-Visual Acquisition of Targets on Touch Phones and Tablets. 2011.

Sean Gustafson, Christian Holz, and Patrick Baudisch. Imaginary Phone: Learning Imaginary Interfaces by Transferring Spatial Memory From a Familiar Device. In *Proc. UIST*, pages 283–292, 2011.

Sean G Gustafson, Bernhard Rabe, and Patrick M Baudisch. Understanding Palm-Based Imaginary Interfaces: The Role of Visual and Tactile Cues when Browsing. In *Proc. CHI*, pages 889–898, 2013.

Taku Hachisu and Masaaki Fukumoto. VacuumTouch: Attractive Force Feedback Interface for Haptic Interactive Surface Using Air Suction. In *Proc. CHI*, pages 411–420, 2014.

Yvonne Jansen, Thorsten Karrer, and Jan Borchers. MudPad: Tactile Feedback and Haptic Texture Overlay for Touch Surfaces. In *Proc. ITS*, pages 11–14, 2010.

S K Kane, J P Bigham, and J O Wobbrock. Slide Rule: Making Mobile Touch Screens Accessible to Blind People Using Multi-Touch Interaction techniques. In *Proc. ASSETS*, pages 73–80, 2008.

S K Kane, M R Morris, A Z Perkins, and D Wigdor. Access Overlays: Improving Non-Visual Access to Large Touch Screens for Blind Users. In *Proc. UIST*, pages 273–282, 2011.

Shaun K Kane, Meredith Ringel Morris, and Jacob O Wobbrock. Touchplates: Low-Cost Tactile Overlays for Visually Impaired Touch Screen Users. In *Proc. ASSETS*, pages 22:1–22:8, 2013.

Robert Kincaid. Tactile Guides for Touch Screen Controls. In *Proc. BCS HCI*, pages 339–344, 2012.

S Landau and L Wells. Merging Tactile Sensory Input and Audio Data by Means of the Talking Tactile Tablet. In *Proc. EuroHaptics*, pages 414–418, 2003.

Vincent Levesque, Louise Oram, Karon MacLean, Andy Cockburn, Nicholas D Marchuk, Dan Johnson, J Edward Colgate, and Michael A Peshkin. Enhancing Physicality in Touch Interaction with Programmable Friction. In *Proc. CHI*, pages 2481–2490, 2011.

David C McCallum and Pourang Irani. ARC-Pad: Absolute+Relative Cursor Positioning for Large Displays with a Mobile Touchscreen. In *Proc. UIST*, pages 153–156, 2009.

Mathieu Nancel, Olivier Chapuis, Emmanuel Pietriga, Xing-Dong Yang, Pourang P Irani, and Michel Beaudouin-Lafon. High-Precision Pointing on Large Wall Displays Using Small Handheld Devices. In *Proc. CHI*, pages 831–840, 2013.

Martin Pielot, Anastasia Kazakova, Tobias Hesselmann, Wilko Heuten, and Susanne Boll. PocketMenu: Non-Visual Menus for Touch Screen Devices. In *Proc. Mobile HCI*, pages 327–330, 2012.

K Pietroszek and E Lank. Clicking Blindly: Using Spatial Correspondence to Select Targets in Multi-Device Environments. In *Proc. MobileHCI*, pages 331–334.

Katie A Siek, Yvonne Rogers, and Kay H Connelly. Fat Finger Worries: How Older and Younger Users Physically Interact with PDAs. In *INTERACT (2005)*, pages 267–280. Springer, January 2005.

Gregg C Vanderheiden. Use of audio-haptic interface techniques to allow nonvisual access to touchscreen appliances. In *Proc. HFES*, volume 40, pages 1266–1266, 1996.

Simon Voelker, Chat Wacharamanotham, and Jan Borchers. An Evaluation of State Switching Methods for Indirect Touch Systems. In *Proc. CHI*, pages 745–754, 2013.

Malte Weiss, Chat Wacharamanotham, Simon Voelker, and Jan Borchers. FingerFlux: Near-Surface Haptic Feedback on Tabletops. In *Proc. UIST*, pages 615–620, 2011.

Daniel Wigdor, Darren Leigh, Clifton Forlines, Samuel Shipman, John Barnwell, Ravin Balakrishnan, and Chia Shen. Under the Table Interaction. In *Proc. UIST*, pages 259–268, 2006.

Daniel Wigdor, Clifton Forlines, Patrick Baudisch, John Barnwell, and Chia Shen. Lucid Touch: A See-Through Mobile Device. In *Proc. UIST*, pages 269–278, 2007.

Jacob O Wobbrock, Brad A Myers, and Htet Htet Aung. The Performance of Hand Postures in Front- and Back-of-Device Interaction for Mobile Computing. *Human-Computer Studies*, 66(12):857–875, December 2008.

Jacob O Wobbrock, Leah Findlater, Darren Gergle, and James J Higgins. The Aligned Rank Transform for Non-parametric Factorial Analyses Using Only Anova Procedures. In *Proc. CHI*, pages 143–146, 2011.

Koji Yatani and Khai Nhut Truong. SemFeel: A User Interface with Semantic Tactile Feedback for Mobile Touch-Screen Devices. In *Proc. UIST*, pages 111–120, 2009.

Shengdong Zhao, Pierre Dragicevic, Mark Chignell, Ravin Balakrishnan, and Patrick Baudisch. Earpod: Eyes-Free Menu Selection using Touch Input and Reactive Audio Feedback. In *Proc. CHI*, pages 1395–1404, 2007.

Index

Typeset November 6, 2014