# Utilising Presence in Places to Support Mobile Interaction

Lauren M. Norrie

Submitted in fulfilment of the requirements for the degree of
*Doctor of Philosophy*

School of Computing Science

College of Science and Engineering
University of Glasgow

May 2016

## Abstract

Physical places are given contextual meaning by the objects and people that make up the space. Presence in physical places can be utilised to support mobile interaction by making access to media and notifications on a smartphone easier and more visible to other people. Smartphone interfaces can be extended into the physical world in a meaningful way by anchoring digital content to artefacts, and interactions situated around physical artefacts can provide contextual meaning to private manipulations with a mobile device. Additionally, places themselves are designed to support a set of tasks, and the logical structure of places can be used to organise content on the smartphone. Menus that adapt the functionality of a smartphone can support the user by presenting the tools most likely to be needed just-in-time, so that information needs can be satisfied quickly and with little cognitive effort. Furthermore, places are often shared with people whom the user knows, and the smartphone can facilitate social situations by providing access to content that stimulates conversation. However, the smartphone can disrupt a collaborative environment, by alerting the user with unimportant notifications, or sucking the user in to the digital world with attractive content that is only shown on a private screen. Sharing smartphone content on a situated display creates an inclusive and unobtrusive user experience, and can increase focus on a primary task by allowing content to be read at a glance.

Mobile interaction situated around artefacts of personal places is investigated as a way to support users to access content from their smartphone while managing their physical presence. A menu that adapts to personal places is evaluated to reduce the time and effort of app navigation, and coordinating smartphone content on a situated display is found to support social engagement and the negotiation of notifications. Improving the sensing of smartphone users in places is a challenge that is out-with the scope of this thesis. Instead, interaction designers and developers should be provided with low-cost positioning tools that utilise presence in places, and enable quantitative and qualitative data to be collected in user evaluations. Two lightweight positioning tools are developed with the low-cost sensors that are currently available: The Microsoft Kinect depth sensor allows movements of a smartphone user to be tracked in a limited area of a place, and Bluetooth beacons enable the larger context of a place to be detected. Positioning experiments with each sensor are performed to highlight the capabilities and limitations of current sensing techniques for designing interactions with a smartphone. Both tools enable prototypes to be built with a rapid prototyping approach, and mobile interactions can be tested with more advanced sensing techniques as they become available.

Sensing technologies are becoming pervasive, and it will soon be possible to perform reliable place detection in-the-wild. Novel interactions that utilise presence in places can support smartphone users by making access to useful functionality easy and more visible to the people who matter most in everyday life.

## Publications

*Virtual Sensors: Rapid Prototyping of Ubiquitous Interaction with a Mobile Phone and a Kinect*, L. Norrie and R. Murray-Smith. In Proc. of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services, Mobile HCI '11, pp. 25 28 (2011).

*Putting Books Back on the Shelf: Situated Interactions with Digital Book Collections on Smartphones*, L. Norrie, M. Koelle, R. Murray-Smith, and M. Kranz. International Conference on Mobile and Ubiquitous Multimedia, MUM '13, (2013).

*Investigating UI Displacements in an Adaptive Mobile Homescreen*, L. Norrie and R. Murray-Smith. International Journal of Mobile Human Computer Interaction, IJMHCI (2015).

*Cast Together: Inclusive and Unobtrusive Mobile Interactions with a Situated Display*, L. Norrie and R. Murray-Smith. International Conference on Pervasive Displays, PerDis '15 (2015).

*Impact of Notification Display Choice on a Typing Task*, L. Norrie and R. Murray-Smith. Workshop on Smarttention, Mobile HCI '15 (2015).

*Notification Display Choice for Smartphone Users: Investigating the Impact of Notification Displays on a Typing Task*, L. Norrie and R. Murray-Smith. International Journal of Mobile Human Computer Interaction, IJMHCI (2016).

**Acknowledgements**

I am grateful to Roderick Murray-Smith for his supervision. His guidance led me through the Ph.D., while providing freedom to find my own research agenda. I thank Matthew Chalmers and his research group for their encouragement to start the Ph.D. and for their help through the initial steps. I also thank those who showed interest in my work and generated fruitful discussions: Andy Cockburn for raising the question of stability on an adaptive homescreen, Leif Azzopardi for demonstrating the value of economic models to compare choice in navigation menus, Stephen Brewster for his advice during the SICSA doctoral consortium, and Matt Jones for his valuable feedback and motivation to continue with this research topic.

During my Ph.D., I had the opportunity to intern at Google Munich and Google London. For this, I thank Stefanie Scherzinger for her guidance, and Björn Bringert for inviting me to return to his team.

Finally, I thank my friends and family, and especially Kiff who has always been very supportive and understanding.

The work carried out in this thesis was funded by EPSRC grants EP/P505534/1 & EP/P504937/1.

**Author's Declaration**

I declare that, except where explicit reference is made to the contribution of others, that this dissertation is the result of my own work and has not been submitted for any other degree at the University of Glasgow or any other institution.

Lauren Norrie

# Table of Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

When attending to information needs on a smartphone, maintaining one's presence in the physical environment is difficult. The rich content and useful functionality provided by applications can attract the fingertips, and more time can be spent looking at the device than intended. Becoming distracted by the smartphone can often be undesirable, especially when concentrating on an activity or sharing a moment with friends. To avoid distraction, one can turn off the smartphone, but this solution is not always possible: While writing a work document, it can be necessary to be aware of alerts and respond to important notifications. Furthermore, the content stored on a smartphone has the potential to facilitate or improve upon everyday life: For example, in a social situation, one might locate an album of digital photos to support a conversation. Despite best intentions, it is difficult to maintain focus on the physical world while performing private interactions with a personal device. Being sucked in to the digital world can not only reduce productivity, but can lead to missing out on special moments with other people. The aim of this thesis is to design mobile interfaces that support access to media and notifications on a smartphone, while enabling the user to maintain their presence in the physical environment. In particular, this thesis focuses on the places that are personal to the user, including rooms of a home, an office or a car. Physical places are given contextual meaning by the objects and people that make up the space. This contextual meaning could be utilised to support mobile interaction, by making access to content on a smartphone easier and more visible to other people.

## 1.1 Supporting Mobile Interaction in Personal Places

Three interaction techniques are explored in this thesis that utilise presence in places to support mobile interaction: situated interactions, adaptive menus and situated displays.


Figure 1.1: Adapting menus to movement around objects and places can make it easier to find content on a smartphone, and can make interactions more visible and inclusive to other people.

### 1.1.1 Situated Interaction

Artefacts of personal places, including photographs on a mantelpiece or a collection of books on a bookshelf, can provide contextual meaning to individuals who frequent the space. Interactions performed around physical artefacts could support the user by making content on a smartphone easier to find. For example, Figure 1.1 (Right) shows a person beside the television set with their smartphone. Explicitly scanning the television with the smartphone could trigger an application to open that can change the programme. Moreover, indicating the device towards the television allows other people to observe the intent to control media on the shared display. Situating interaction around physical artefacts moves the focus of attention away from the smartphone and into a place, and this could make interactions with a private device more visible and inclusive to other people.

### 1.1.2 Adaptive Menus

Personal places are usually designed to support a set of tasks: For example, a kitchen should support the preparation of food, where as a bedroom should be comfortable for sleeping in. On a smartphone, digital tasks are completed with applications. Menus that adapt to the contextual meaning of a place could make apps easier to navigate and reduce the time spent attending to information needs. When the smartphone detects the user to be in a kitchen, apps that support a cooking task could become easier to launch. Alarm clock and book reading apps that support the user in waking up or going to sleep could be prioritised in the bedroom. Furthermore, app launch habits could be made transparent to the user, so as to encourage self-reflection and positive behavioural change. For example, one might reflect on the accumulative time spent attending to work email at home, and change this behaviour to focus more on family life. Adaptive menus could make the navigation of apps more efficient, and have the potential to increase awareness of smartphone habits in places.


Figure 1.2: Interactions with private displays can make it difficult to maintain physical presence.

### 1.1.3 Shared Displays

Oftentimes, personal places are shared with people whom the user knows. Figure 1.2 (Right) captures two people who are visually focused on their smartphones, and the social context is fragmented by the content that each person can see on their personal device. In contrast, Figure 1.3 shows a group of people gathered around a television set. In this scenario, group members are invited to see any information that is shown on the communal display, and the content can encourage social engagement. Shared displays could be utilised to support social engagement in personal places by coordinating content from a smartphone, and allow notifications to be read at a glance. Furthermore, presenting smartphone notifications close to the social task could reduce disruption, especially when a notification can be ignored.

## 1.2 Contributions

1. An app launch and notification dataset collected in the context of personal places provides quantitative evidence that app use can be linked to places. A technology-driven approach made it possible to collect this dataset. An interactive visualisation was created to explore the dataset and gain insights into app use in places. In a user experiment, evidence was found to support a rank order of apps when movements in the adaptive homescreen menu are large, and an alphabetical order when the adaptive menu is stable. Feedback from a user study with this adaptive menu indicates that adapting to the context of places made it easier to access apps that are frequently used there.

2. A probe was developed for sharing media and notifications from a smartphone on a situated display. Results from a user evaluation with this probe during a social board game task provides evidence that sharing smartphone media with a co-located group can stimulate conversations, while reducing direct interaction with a private display.

Figure 1.3: A communal display can allow other people to share a social context of what is shown on the screen, without the fragmentation of private displays. (The image of the family watching the television is available with a Creative Commons licence[1].)

Results from a user evaluation of notification displays with an individual typing task provides evidence that a faster approach to reading a notification is preferred while concentrating on a primary task, and is more important than a faster method of responding to a notification. In particular, glanceable notification displays that were close to the typing task (desktop PC, smartwatch, situated display) were the preferred way to read notifications, and reduced the time to ignore a notification when a response was not required.

3. Prototypes were developed that utilise the relationship between physical artefacts and digital content. Each prototype enabled user evaluations to be performed that explore interaction around physical objects, structures and places: bookmarking web pages with objects, organising digital books around a physical bookshelf, and adapting a menu of smartphone apps to physical places. Situating interactions in places was found to reduce visual attention to a private display. Results from these evaluation highlight the opportunities of personal places to organise smartphone content, and increase the visibility of private smartphone interactions for other people.

4. Two rapid-prototyping tools contribute a low-cost approach to sensing the mobile user environment with sensors that are currently available: the Microsoft Kinect was demonstrated to track movements of a smartphone user in a small area, and Bluetooth beacons enable the context of logical places to be detected. Each tool was designed

[1] https://commons.wikimedia.org/wiki/File:Family_watching_television_1958.jpg

to enable rapid exploration of smartphone interactions in personal environments, and enabled user studies to be performed with working prototypes. Results of positioning experiments with each tool highlight the capabilities and limitations of these sensing techniques for designing smartphone interactions.

## 1.3 Outline

The remainder of this thesis is organised in the following chapters:

**Chapter 2.** A review of relevant literature in the field of human computer interaction, with a focus on the opportunities of situated interaction, adaptive menus and shared displays, in addition to existing approaches to sensing the user environment.

**Chapter 3.** A review of the research methods and key research questions. The technology-driven research approach is explained, along with the ethics procedure used in the user experiments. The limitations of the research approach and the generalisability of the results are also highlighted.

**Chapter 4.** Chapter 4 presents two rapid prototyping tools: one tool utilises the Microsoft Kinect depth sensor to track the movement of a smartphone user in the limited area of a place, and another relates Bluetooth beacons to logical labels to enable the wider context of personal places to be detected. Positioning experiments are performed with each tool to highlight the capabilities and limitations of these two low-cost sensing technologies.

**Chapter 5.** An initial exploration of interacting with the architectural environment. Existing bonds between people and their personal places are considered in a questionnaire, and the potential to utilise these relationships to supplement interaction with a digital interface are indicated in the results. Three prototypes are developed, and user evaluations are conducted to demonstrate the opportunities of each approach: bookmarking websites with tagged physical objects, browsing digital book collections around physical structures, and adapting mobile apps on the homescreen to physical places.

**Chapter 6.** A history of app launches is gathered in the context of places, including a personal car, office and rooms in a home, and is used to adapt the mobile homescreen to show only the apps that a user needs when located in a place. Evaluations conducted with the adaptive homescreen indicate opportunities to support app navigation in personal places.

**Chapter 7.** A situated interface is designed to share the smartphone events and personal profiles of nearby users, and acts as a probe for inclusive and unobtrusive mobile interactions. Evaluations are conducted with the situated display application, and the results demonstrate the potential of coordinating personal preferences automatically to stimulate conversation without active engagement with a mobile device.

**Chapter 8.** Contributions and implications for design are summarised with opportunities for future work.

## 1.4 Timeline

A timeline is provided of the work carried out in this thesis, along with the release of technologies and attendance at events that shaped and inspired the direction of this thesis:

### 2010

- November: Microsoft Kinect released.

### 2011

- **February: Submitted paper on Virtual Sensors [112] (Section 4.1).**
- **April: Submitted workshop paper on Interacting with Multiple Mobile Devices using the Kinect [111].**
- *June - October: Internship at Google Munich.*
- October: Proximity Toolkit [105] published.
- **November - December: Collaboration with visiting Masters student on Situated Interactions with Digital Book Collections on Smartphones [110] (Section 5.3).**

### 2012

- **September: Sent questionnaire on relating websites to physical objects (Section 5.1).**
- **September: Conducted experiment on relating websites to physical objects (Section 5.2).**
- *October - February 2013: Internship at Google London.*

## 2013

- June: Announcement of iBeacon protocol for Bluetooth Low-Energy (BLE) beacons.
- **August: Conducted initial experiment on app tracking in personal places (Section 5.4).**
- **September: Conducted experiment on stability on the adaptive homescreen (Section 6.1).**

## 2014

- April: Collaboration with Masters student on shared notification displays [43].
- **October: Conducted experiment on app tracking in personal places (Section 6.2).**

## 2015

- **January: Conducted user study to evaluate Cast Together in Places (Section 7.2).**
- **February: Conducted experiment on Impact of Notification Display Choice on a Typing Task (Section 7.3).**
- **May: Conducted user experience evaluation of Cast Together (Section 7.1).**

# Chapter 2

# Background

The physical environment can provide context to situate information and interact around physical objects and structures. This chapter reviews literature in the field of human computer interaction, with a focus on situated interaction, and opportunities to utilise presence in places with adaptive and situated interfaces. Existing approaches to sensing the user environment are also reviewed.

## 2.1 Situated Interactions

### 2.1.1 Bridging the Gap

Bridging the gap between physical and digital artefacts can be related to coupling 'bits and atoms' [83]. Fitzmaurice envisioned situated information spaces as a way to anchor digital information to physical objects, such that it may be browsed and manipulated in the context in which it originated [54]. The Chameleon, described in [53], is an early example of a spatially-aware mobile interface, which enables a palmtop computer to 'act as an information lens near physical objects'. For example, to display weather information while scanning a physical map. The concept of situated information spaces could be extended to consider personal information stored on a smartphone.

A reactive environment is described in [32], and the key features recommend technology to be invisible to the user, and for a system to communicate the state of its non-visual interfaces to the user with sufficient feedback. Additionally, interfaces in a reactive environment are recommended to adapt to the preferences of its users, and controls should be provided to manually override the system. The design guidelines for a reactive environment provide a starting point to design interfaces that react to the context of personal places.

### 2.1.2 Interaction Boundaries

In [72], challenges of interaction design are highlighted in an age where sensors are embedded in the environment, including how to interact with technologies that are not necessarily visible, and how to define where the boundaries of control affect the users of a space. In [14], Billinghurst *et al.* classify exocentric spatial information displays as world-stabilised, since the information displayed to the user changes with respect to position and orientation. Bearing-based interaction demonstrates pointing as way of interacting with exocentric geo-coded information in [142], and BodySpace [143] demonstrates egocentric interaction with a media application. BodySpace utilises areas of the body as mnemonics for functions that are controlled by gesturing the device in each area.

### 2.1.3 Manipulations and Effects

In [126], a taxonomy for spectators and performers of interactive interfaces is defined. Manipulations and their effects can range from expressive (revealing manipulations and effects), secretive (hidden manipulations and effects), magical (hidden manipulations and revealing effects) or suspenseful (revealing manipulations and hidden effects). It can be argued under this definition that smartphone interactions are secretive, since the manipulations and effects on the screen are hidden to those nearby. In comparison, situating interaction in the physical environment could create manipulations that are revealed to others, and could either be expressive or suspenseful: If the effect is displayed on a shared display then a situated interaction could be expressive, or if the effect is displayed on the private smartphone display then an effect might be suspenseful. However, as movement around the physical environment does not necessarily involve interaction with a smartphone, it could also be possible to create interactions that appear either magical or secretive.

## 2.2 Adaptive Menus

### 2.2.1 Menu Navigation

Menu interfaces support smartphone users to navigate content and satisfy digital information needs. However, the time to navigate a menu interface impacts the time to resume an activity or social situation. Split Menus, designed in [135], demonstrate that menu selection time can be decreased by moving or copying the top 4 frequently used items to the top of a menu in a desktop application. A static split menu is optimised with the most frequently used items at the time that the menu is created, after which no more updates are applied. In comparison, an adaptive split menu reacts to usage history over time. Adaptable menus allow the user to take control and make manual adjustments to the inclusion of items in the menu. For example, the homescreen on an Android smartphone is an adaptable interface, as users have control over the selection and placement of apps in the homescreen panels. Static, adaptive and adaptable split menus are compared in [49]. The naturally generated data of a single MS Office user was used to populate the adaptive menu, and it was found that the majority of participants wanted a personalised menu (adaptive or adaptable). It was also found that participants were better at customising an adaptable interface after they had used the adaptive one. This suggests that an adaptive component could support users with organising their app icons. In [57], it is noted that adaptations can be more appropriate for novice users. The impact of screen size in adaptive user interfaces was investigated in [50], and it was found that an 'adaptive interface is more beneficial when screen real estate is constrained' and that 'adaptive interfaces are low risk for small screens' (p. 1254). The potential benefits of adaptive interfaces on small screen devices provides motivation for exploration on the smartphone.

### 2.2.2 Digital Information Needs

In [71], it is argued that everyday interactions should be understood so that digital support can be designed around them. Similarly, in [77] it is found that few mobile Internet needs are location specific and suggest that future mobile information services consider a wider context of use, including social interactions and situated activities. This motivates the exploration of information needs in personal places. Individuals interact with their mobile devices in a diverse set of ways [47, 164]. By using the mobile phone as a wearable sensor, it is possible to identify patterns in user behaviour [44]. In [85], data is collected on smartphone users and it was found that a majority of active smartphone use is in the home. Diary studies provide insight into user behaviour in more find-grained places [28, 30, 38, 141, 77, 153], including, Home, Office, 'On the move', 'Other meaningful', Elsewhere and Abroad [153, 28]. Detecting mobile information needs in personal places could be supported with an automated approach.

### 2.2.3 Context

The definition of context is core to the discussion of ubiquitous computing. Weiser imagined that user activity will be inferred from sensors embedded in the physical environment [159]. However, as humans are highly unpredictable, detecting activity is a difficult challenge [128]. In [42], it is argued that context itself cannot be modelled and suggests that everyday action should be focused on as a way of understanding the user environment. Dey provides a definition of context that can be applied by interaction designers in [39]:

> ‘Context is any information that can be used to characterise the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves... A system is context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user’s task.’

### 2.2.4 Adaptive Homescreen Menus

With an understanding of information needs in places, it is possible to adapt the mobile interface. The homescreen is the main interface of a smartphone, and the most common way of navigating apps on a smartphone, as found in [66] in their study of app launching habits. In [16], it was found that users spend an average of 59.23 minutes per day on their device, with app use spread intermittently throughout the day. This presents a different use case to a desktop application, where the system is used in concentrated periods. Furthermore, apps can be installed and uninstalled on a smartphone, changing the range of functions that can be displayed over time [138], similar to a desktop PC. In comparison to a desktop PC, mobile devices are used in a variety of contexts, and this also affects the apps that are likely to be used [16]. To make accurate predictions, the adaptive model must update frequently to keep up with the continuously changing context. Therefore, it is important to consider the design of an adaptive menu in a mobile context, and understand how frequent adaptations, and unnecessary changes, will affect usability.

### 2.2.5 Automation

Humans become tired or bored when they are no longer actively involved in a process: When given a monitoring task, they can lose track of the context of the system that they are expected to supervise [137]. Instead, human and machine should work together. Computation should reduce cognitive load, while allowing the user to take part in the tasks that they enjoy. As Knuth states in [88], ‘One of the best ways to keep up the spirits of a system user is to provide routines that he can interact with... Some tasks are best done by machine, while others are best done by human insight; and a properly designed system will find the right balance’. Therefore, it is important to evaluate the impact of an adaptive system with users early in the design process.

### 2.2.6 Negotiated Interaction

The H-Metaphor, described in [55], is a negotiated interaction technique that relates the control of a system to holding the reins of a horse: users can influence an automated process in order to converge on a solution to a problem. This metaphor is also reflected in mixed-initiative systems that interweave direct control with automation [80]. Negotiated interaction deals with the uncertainty of context and human behaviour by allowing human initiative to guide the system when appropriate. When human input is not provided, the goals of interaction can be predicted by historical models of user activity or other algorithmically defined patterns. Systems that learn user behaviour can predict future actions. For example, Just-in-time interaction predicts the current goals of a user from their context and anticipates relevant information as it is required, without the need to form a query [23]. Similarly, the appropriate feedback modalities for a mobile device can be learned in the context of geographic locations [41]. Negotiated interactions let the user influence a context-aware system, and should be considered in the design of menus that adapts as the user moves between places.

### 2.2.7 Context-Aware Recommendation Systems

Context-aware recommendation systems are built on an understanding mobile information needs [136, 16, 37]. In [16], a dataset of application usage was gathered on Android mobile phones from 4100 users of their Appazaar context-aware recommender application. They analysed the app usage in terms of chains of application usage and also in terms of context, namely time and geographic location, which helped to improve the prediction accuracy. Similarly, Applause learns the location of app installs to recommend apps for other users to install in a given location [37]. In [1, 84, 168, 15], techniques and challenges are discussed for designing context-aware recommender systems. Context-aware recommendation systems could benefit from the presence of co-located people in a personal place.

### 2.2.8 Self-Reflection

Self-reflection applications encourage users to quantify themselves by collecting long-term data. Persuasive applications build on this data to motivate behavioural change. For example, logging affective state can be used to adapt an intelligent user interface [145]. Life-logging applications have requirements for long-term archiving, privacy and efficiency [125]. [94] identify reasons behind self-reflection and what users aim to understand from personal data. With a better understanding of ones own habits, it becomes easier to change behaviours. For example, in [98], AppDetox provides a service for user to actively disable mobile apps, to discourage their use when they wish to focus.

Self-reflection can encourage social awareness and co-located engagement. [33] demonstrate that displaying social network posts by family members on a situated display provides ambient awareness and strengthens connectedness and in-person encounters. In [19], a trivia application is demonstrated that can help colleagues learn about each other, and can encourage social engagement in meetings by providing discussion points that facilitate the social situation. Co-ordinating social media and events on a situated display could stimulate conversation in personal places, without active engagement with a smartphone.

## 2.3 Shared Displays

### 2.3.1 Interruptions

Notification alerts can draw multiple people away from a social situation to engage with a private display ('collateral disruption') [73]. In [107], McFarlane identifies four strategies of managing interruptions: immediate, scheduled (defined intervals), negotiated (user determined), and mediated (third party decides). In [106], each approach to coordinating interruptions are compared, and it was found that when people are forced to take immediate action, interruption tasks are completed quickly but more mistakes are made in the primary task, and more task switches are involved. In contrast, people perform very well when they can negotiate interruptions themselves, but providing control over the onset of an interruption will increase the time until the interruption task is attended to. This result motivates notification displays that support the negotiation of interruptions, and allow users to better manage the disruption to a primary task.

### 2.3.2 Collaborative Media Sharing

A visual attention switch [124] to a private screen creates a barrier between smartphone users and other co-located persons, and can be caused by a notification alert or a need to access device functionality by navigating a menu interface. Sharing media from a smartphone can help to satisfy digital information needs while maintaining focus on a social situation. Co-located photo sharing can be supported by placing mobile devices together to create a larger screen [99], and this type of adhoc utilisation of devices can scale with little infrastructure or manual configuration [134]. However, as more people are invited to share photos on the display, there is a challenge of how to coordinate the sharing of media.

Figure 2.1: A low-resolution, shared notification display, simulated on an Android tablet.

A floor policy defines how users request and grant access to control a shared multimedia application [35]. A request can be made by an inactive user by taking a user action (implicit request) or by asking the active user for control (explicit request). The active user can grant control by accepting or denying a request (explicit grant) or a request can always be granted with a user action eventually taking effect (implicit grant). No floor policy means that all users can take action simultaneously (free-for-all).

In [2], an experiment with three floor policies is described that synchronises the displays of co-located mobile devices. Explicit, implicit and free-for-all floor policies were compared to control the shared application (e.g. change the photo, draw, zoom). It was found that requiring users to explicitly request and grant the host-token encouraged the most storytelling, and was the preferred mode as it left little ambiguity of who was in control of the display. An implicit mode, on the other hand, was ambiguous about who had control of the display, and participants frequently interrupted the storytelling to make requests verbally. The free-for-all mode was the most chaotic, and storytelling was abandoned in favour of taking control as much as possible. A limitation of this approach is that the preferences of only a single user are included at a time.

In [117], a shared music system (Jukola) is presented that takes an inclusive approach to coordinate media preferences. Co-located smartphone users can become involved in the selection of music by actively voting on upcoming songs with their personal device. The overall preference of the democratic group is satisfied by the system, which ranks songs according to their votes. This result can be build on by using the music profiles stored on a personal device to implicitly share preferences, without active engagement.

### 2.3.3 Notification Displays

The lack of screen space on mobile devices can be overcome with multi-modal techniques, such as presenting buttons with tactile and audio modalities [21]. However, audio and tactile modalities convey limited information and are not always appropriate [68, 78].

In [68], mobile notification systems are classified according to subtleness and publicity. It is considered desirable for notifications to be both subtle and public to allow co-located people to be aware of the interaction, without creating a disturbance. As private notifications are hidden to people nearby, this can increase the risk of misinterpretation of a user's reaction; in comparison, a public notification is transparent, and allows others to understand a response to a notification. Under this classification, auditory cues are considered public and intrusive, whereas tactile cues are subtle and private.

The Wearable Remembrance Agent, described in [127], is head mounted display that is considered to be both private and intrusive, since only the user can view the notifications but wearing a head mounted display can be seen as distracting to other people. In [100], a study is performed with the NotifEye smart eyewear system, and it was found that participants were able to negotiate smartphone notifications while walking in public with a subtle input device worn on the finger. It was also found that the novel eyewear drew attention from passers-by.

Notification displays that are considered to be subtle and public include: Active Wallpaper [163], the Pinwheels [36], the Dangling String [160] and the Reminder Bracelet [67]. A smartwatch could also be considered as a subtle and public display since the wrist is visible to nearby people. In [119], smartwatches are explicitly used to present the state of a user to others. A limitation of a smartwatch is that individual smartphone users require additional hardware at the expense of the user. Work on subtle and public notifications can be extended to consider a shared situated display.

In [43], a low-resolution display was designed, as displayed in Figure 2.1. Thumbnail images associated with a notification appear in low-resolution on an Android tablet. Though the low-resolution display accounts for privacy associated with notifications, there is a lack of accountability and information associated with the images. Furthermore, only a single notification can appear on the display at once, making it difficult for users to know if a notification has been missed. The design of a shared notification display could be improved upon to persist notifications, and provide accountability to the smartphone user for whom the notification is intended.

### 2.3.4 Shared Information Systems

In [97], it was found that smartphone users experienced stress reactions when highly personal information was shared on a public display compared to a handheld device. Therefore, when sharing personal user content with a shared display, it is important for the system design to consider when it makes sense for a user to be identified. In [131], personalisation is separated into three levels: personalised information that must not be shown in public; that can be shown in public; and information that can be shown if no link to the initiator can be drawn.

The privacy of personal information can be controlled on a mobile interface by colour coding information, hiding information behind blinders that reveal when they are explicitly touched, or adapting hidden information to the person in view [62, 148]. However, sharing information on a situated display makes personal information public. It is possible to control the level of information on a situated display with proxemic interaction [7]. Similarly, adapting to presence in a place provides a potential solution to manage the privacy of a shared information system in personal places.

### 2.3.5 Intelligibility and Accountability

Systems that react to a contextual model act with uncertainty. In [25], it is argued that users should be allowed to interpret and influence the system's understanding of context by exposing the 'seams' of a sensing space. In [11], both intelligibility and accountability are argued as necessary to accommodate for the human and social aspects of context-aware systems. Intelligibility respects human initiative by making the internal state of the system visible to the user, and provides controls to act on this state, to allow the user to decide if the desired behaviour is being recognised. Accountability respects interpersonal actions by allowing individuals to understand how they are affected by the context aware system, through identification of user and system actions. In [95], it was found that intelligibility is helpful for applications with high certainty; for applications with low certainty, intelligibility can help users appreciate and forgive applications if they behave inappropriately, at the risk of users losing faith in the system's ability. It is recommended that intelligibility is included in the system design when an acceptably high accuracy is achieved.


(a) SHAKE SK6, SHAKE SK7, JAKE and USB Bluetooth beacons.

(b) Microsoft Kinect depth sensor.

(c) NFC tags.

Figure 2.2: Examples of (a) Bluetooth, (b) Motion, and (c) NFC sensors.

## 2.4 Sensing the User Environment

### 2.4.1 Positioning

#### Global Positioning System (GPS)

Location technology has revolutionised the way in which users interact with mobile devices. With a mobile phone, a user can easily know where they are and how to navigate an unfamiliar route, and search results can be returned in order of geographic proximity to the user. Location-aware applications use the Global Positioning System (GPS) to estimate the location of a user relative to the world. The GPS system estimates position based on the round trip time of a message that is sent to a GPS satellite. The limitations of the GPS system is that signals are weakened by reflections in the environment and so the position estimate is only accurate to around 20 meters and operates outdoors [27]. Compared to GPS, indoor positioning is still a technological challenge and no standard exists that estimates this reliably [90].

#### Ultrasonic Positioning System

ActiveBat, described in [123], is an early example of a ultrasonic positioning system, which requires a central radio transmitter to be embedded in the ceiling. The transmitter sends request packets to which an ultrasonic component responds, and the receiver measures the time interval between a sent packet and pulse response to estimate the distance from the central unit, and can be interpreted as the position in the room. In [121], Cricket is presented: a decentralised ultrasonic system with radio frequency that helps devices learn their position rather than explicitly tracking users. The drawback of this approach is that the required technology is not widely available.

### Wi-Fi Positioning

Wi-Fi technologies have also been used to perform positioning. A database of wireless networks and their signal strength 'fingerprint' at sampled positions of a room can be used to estimate the position based on a current reading [56]. Though wireless networks are more widespread in current society compared to ultrasonic transmitters, the fingerprinting technique is time consuming to perform and there is still work to be done to improve the reliability of Wi-Fi positioning systems [152, 147]. Furthermore, little data has been collected about positions in personal places, and more work is required to enable end users to set up a Wi-Fi positioning system.

### Bluetooth Positioning

Similar to Wi-Fi positioning, Bluetooth signals can be used to detect positions indoors. Traditionally, Bluetooth was designed for communication and is used by many devices, including wireless headsets, keyboards and mobile phones. Figure 4.6 (a) displays a set of traditional sensors that can be used as Bluetooth beacons. When performing localisation with Bluetooth, the placement of beacons should first be considered [26]. Detecting the nearest Bluetooth beacon can provide a high-level classification of the room-level location [9]. In [40], Bluetooth was used to investigate the proximity of smartphone users to their device, and show that it is possible to detect when smartphone users have their devices at arm + room level with 90% accuracy. A limitation of traditional Bluetooth beacons is the requirement of an external power source, and lack of standardisation to use the signals for positioning.

In 2014, Low-Energy Bluetooth (LE) beacons became popular for marking positions indoors, which are based on the Apple iBeacon protocol.[1] The beacons have a shorter communication range than traditional Bluetooth, and last several months or years on a single battery, and therefore can be self-contained units. Kontakt.io were one of the first manufacturers to adopt the iBeacon standard. An example of Kontakt.io beacons is displayed in Figure 4.6 (b). Applications with Bluetooth LE beacons have been explored for commerce. For example, a supermarket might put a Bluetooth beacon next to a special promotion, and nearby shoppers with the appropriate app installed can be notified just-in-time. LE Beacons emit a short-range signal that a receiver can detect, and estimate proximity to a position in the physical environment.

More recently, Google have developed addressable Bluetooth LE beacons for the Physical Web[2], which aims to encourage the development of locally relevant web services that are linked to physical places, and that can be explored with a web browser. This approach will

[1] `https://developer.apple.com/ibeacon`

[2] `https://google.github.io/physical-web/`

reduce the need to download an app for each service, and will increase the utility of many small but useful applications.

An inherent limitation of radio signals is that they are affected by interference. This makes it difficult to reliably estimate the distance to a receiver. Bluetooth signals can be fused with Wi-Fi signals to improve reliability [156, 10]. The process of configuring a database is also an undesirable overhead for applications that require only the resultant position [29]. There is a requirement to make indoor positioning systems quick and simple to deploy to explore real-world applications that use this context [104, 76, 22].

### 2.4.2 Visual Tracking

An alternative to detecting the position of a user relative to a beacon is to perform visual tracking. Hand tracking is possible by tracking an LED with a web-camera [155], which is also demonstrated with the Sony Move controller. The Nintendo Wii controller tracks the hand with Infra-red [161]. With these technologies, it is possible to design interactions that involve pointing [3, 122]. However, a limitation is that users are required to hold a controller.

Computer vision can track the whole body of a user. Motion tracking has been achieved with markers and a 2D camera [139], and 3D depth cameras provide a marker-less approach [24]. Marker-less tracking has the advantage of being unaffected by varied lighting conditions [140], and occluded markers. Furthermore, 3D tracking can be performed without holding any equipment. The limitation of vision-based techniques is that a user must be in view of the camera, and limits interaction to the camera's field-of-view and line-of-sight. Successfully fusing multiple depth-sensors can improve coverage of vision-based positioning, and reduce the uncertainty of detecting users [24]. The Microsoft Kinect can be used to mediate pointing with a customer in a retail environment, without holding a controller [60]. More recently, Google Project Tango[3] demonstrates that 3D depth sensing could be available in a mobile device.

### 2.4.3 Tagging

A limitation of tracking objects with computer vision is that the position of the object is relative to the camera. By tagging objects with proximity sensors, digital information can be explored relative to the object. Indoor-position can be estimated with active RFID [109] and Near Field Communication (NFC) [157]. NFC tags, displayed in Figure 4.6 (c), can be used to create tagged displays [69], and digital content can be retrieved by scanning them with an NFC-enabled device. Both static and dynamic displays are possible, such as posters and

[3]https://www.google.com/atap/project-tango/

projections. This has been demonstrated in a study where physical objects could be tagged with messages, and scanning the messages would automatically post to a social network [70]. A limitation of object tagging is that a number of tags should be managed by the user, and the discoverability of tagged content is an important design question. In [65], an icon set is presented that describes tasks that can be performed with a tagged display.

### 2.4.4 Rapid Prototyping

Rapid prototyping is the process of simulating software design ideas quickly [146]. Until sensing techniques become more readily available, it is desirable to take a rapid-prototyping approach when designing interactions. Amarino is a rapid prototyping framework for ubicomp systems based on Android and the Arduino microcontroller [86]. A limitation is that many microcontrollers can be required to simulate complex interactions. In [20], the GAIM prototyping framework is presented that abstracts the input device by adapting to the dynamic availability of input hardware, in order to ease the development of multi-platform games. The Digital Replay System presented in [61] allows multiple data sources to be recorded and replayed to aid the evaluation of interactive applications. In [105], the Proximity Toolkit is presented that provides extensive support for detecting the proximity of users in a place.

## 2.5 Summary

This chapter focused on the barriers of mobile interaction in places, the opportunities of adaptive interfaces and situated information, and existing approaches to sensing the user environment.

Information stored on a smartphone can be situated around artefacts of a place, and has the potential to make interactions visible to observers. A challenge of situating information is how to make the situated interfaces discoverable to the user and to other people, and how to design and evaluate interactions with the available sensing technology.

Adaptive menus can support the user by displaying the applications that a user might wish to launch just-in-time. However, adaptations can be difficult for users to follow, and ways to manage the stability of an adaptive menu should be explored.

Sharing media with co-located persons on a situated display could create an inclusive user experience, and has the potential to encourage self-reflection on how oneself is represented in the digital world. Presenting smartphone notifications on a situated display could also increase focus on a primary task by allowing content to be read at a glance. When sharing content from a personal device with others, privacy issues must be considered, and users should be able to control the content that they share, and should be held accountable for their actions.

Improving the sensing of smartphone users in places is a challenge that is out-with the scope of this thesis. There are many limitations with current indoor sensing techniques, and lightweight position sensing is required to design interactions that utilise presence in physical environments. Tools should be designed that enable prototypes to be tested with more advanced sensing techniques when they become available.

The following chapter will discuss challenges and opportunities to build on, and the necessary research approach.

# Chapter 3

# Research Methods

Previous work highlights the challenges of sensing the user environment, and guidelines to consider when designing interactions with context-aware interfaces. This chapter presents the research questions that are tackled by this thesis, and the associated challenges motivate the research approach. Existing approaches to rapid prototyping and sensing the user environment are reviewed, and the technology-driven research approach that is used to investigate the research questions is explained, along with the participants and ethics procedure used in the experiments. Limitations of the research approach are also highlighted along with the generalisability of the results.

## 3.1 Research Questions

This thesis explores four research questions, with the aim to support mobile interaction in personal places:

1. How might mobile interaction be situated around artefacts of personal places, and support users to access content from their smartphone while managing their physical presence?
2. How might adapting menus to personal places reduce the time and effort of app navigation on the smartphone, and increase self-reflection on where apps are used?
3. How might coordinating smartphone content on a situated display support social engagement and the negotiation of notifications?
4. What are the capabilities and limitations of a rapid-prototyping approach with the Microsoft Kinect depth sensor and Bluetooth beacons to detect the smartphone in places?

## 3.2 Prototyping

Human Computer Interaction studies the interaction between people and designed systems, and is multidisciplinary field that draws from computing science, human factors, social science and design. Psychology, anthropology and sociology are all relevant to the way that humans interact with computers. However, the process of designing physical artefacts differs to the scientific model. Designers and engineers iteratively produce working prototypes based on guidelines to develop a final product. In [101], it is argued to be necessary to triangulate across scientific and design disciplines to produce results that are significantly more robust and useful. Social science research requires trade-offs to be made, as it is not possible to addresses all the possible threats to experiment validity: generalised populations, precisely controlled and measured variables related to the behaviour of interest and observed behaviours in real environments. Furthermore, the interaction between people and technology is co-adaptive, and it is necessary to evaluate working prototypes in different environments. Through triangulation, it is possible to seek convergence or divergence, by comparing results from controlled experiments with studies in dynamic environments.

Low-fidelity prototyping, including throwaway paper interfaces and wizard-of-oz studies, are inexpensive techniques for the rapid exploration of design ideas. Paper interfaces enable the designer to sketch multiple flows, and step through critical paths with users, allowing designs of the user interface to be iterated on quickly. Wizard-of-oz studies are controlled by a human operator who mocks the response to user interactions, and enables rapid feedback on the performance of novel interaction techniques without the requirement of a reliable system behaviour. Low-fidelity prototyping is valuable in the initial design stages to gain feedback on core flows and interaction techniques. However, low-fidelity are highly involving for the human operators, and therefore are usually tested with limited functionality. In comparison, technology-driven prototyping enables interactions to be evaluated with working systems, and quantitative data can be collected over long periods of time.

This thesis explores lightweight position sensing for rapid-prototyping with both the Microsoft Kinect and Bluetooth beacons. The Proximity Toolkit[1] was released during this thesis work, and has since been extended to support technologies for rapid prototyping of proxemic-aware systems, including the Microsoft Kinect. Lightweight tools for technology-driven prototyping will enable developers and interaction designers to explore novel interaction with the smartphone, and gather both quantitative and qualitative feedback from users in many contexts and over long periods of time. By taking a sensor-agnostic approach to prototyping, interactions that are designed with a technology-driven approach utilise more advanced sensing techniques as they become available.

[1] `http://grouplab.cpsc.ucalgary.ca/cookbook/index.php/Toolkits/ProximityToolkit`

## 3.3 Methodology

The goal of this thesis is to evaluate smartphone interactions that utilise presence in places, with an aim to support access to content while managing attention to an activity or social situation. Three interaction techniques are explored, along with a rapid-prototyping approach to detecting the user environment. Lightweight tools for position sensing are developed and used to track the movement of a smartphone user in a room and detect the contextual meaning of a place. A technology-driven approach is taken to evaluate interactions, by developing working prototypes and designing interactions with lightweight position sensing tools. Situated interactions in places are explored by using the Microsoft Kinect to tag individual objects and structures with digital content. An adaptive menu of smartphone smartphone applications is developed on the homescreen and reacts to logical places by using Bluetooth beacons to mark landmarks. A situated display that shares content and notifications from a smartphone is also explored, and is designed with adaptive content and situated interactions, acting as a probe for inclusive and unobtrusive interactions. The insights gained from user evaluations conducted with each prototype provide the main contributions of this thesis.

All user studies were reviewed by the University of Glasgow ethics committee. This process required evaluations to be prepared in advance. The review process took several weeks or months from submitting an application depending on the schedule of the committee. The application specified the description of the evaluation, the participant recruitment process and any compensation provided to participants. All evaluations performed in this thesis, with the exception of Section 7.3, recruited participants on a voluntary basis and without monetary compensation. The participant information sheets and consent forms were also reviewed by the committee, and the signed paper copies are held by the University of Glasgow.

## 3.4 Limitations and Generalisability

The quantitative data gathered in this thesis is limited by the low-cost sensing techniques that were available. For example, in Section 5.2, it was not possible to collect data of one participant due to issues with the Kinect tracking the user behind a table. Detecting places with traditional Bluetooth beacons was also unreliable, as demonstrated with the Appwhere prototype in Section 5.4. However, as prototypes were developed with a sensor-agnostic approach, new sensing techniques could be swapped in when they became available. For example, the My Places tool was improved with the new Low-Energy Bluetooth beacons, and the Appwhere prototype in Section 6.2 continued to operate without any changes. As the sensing techniques become more ubiquitous in everyday life, it will become easier for future researchers to test how well these findings generalise to a wider population of smartphone users.

This thesis focuses on places that are personal to the user, including a personal office, car or rooms in a home. An assumption is made that personal places are shared by people who the user knows, which separates this work from public displays and location-aware services, as strangers might also share or visit a public place. However, real-life contexts can be complicated, and the privacy issues associated with making personal information public is still a challenge that needs to be faced before sharing content from a smartphone can generalise to every day life.

The results in this thesis are also limited to a small population of participants and places. The number of participants used in the evaluations range from 3 participants in Section 4.1 to 30 participants in Section 7.3. Many of the participants used in this thesis were recruited as students from the University of Glasgow School of Computing Science, with the exception of Sections 5.4, 6.2, 7.1 and 7.2: In these studies, participants were known to the author, and some participants took part in multiple experiments, as noted in the experiment description. Though the relationship of these participants to the author is a confounding factor for the findings of these experiments, the rapid feedback from peers was valuable to iterate on the design of the adaptive homescreen and situated display prototypes. In addition to feedback recorded in questionnaires and quantitative data gathered with each prototype, it was valuable to observe how these prototypes were used in the wild, and triangulate between the results. However, future work is required to evaluate these prototypes with a more diverse population of users to validate their generalisability beyond technical students and the social contacts of the author.

## 3.5 Summary

This chapter described the research methods used in this thesis. The research questions were presented, and build on the challenges and opportunities found in previous literature. A technology-driven research approach was reasoned to be necessary to gain insights from multiple perspectives, through the design of working prototypes, and the refinement of insights gained from controlled lab experiments and observational user studies. The next chapter will present the rapid-prototyping tools that sense the user environment, and are used in the remainder of the thesis to develop place-aware mobile interfaces.

# Chapter 4

# Lightweight Position Sensing for Rapid Prototyping

This chapter presents two approaches to lightweight position sensing that utilise the capabilities of 3D depth sensing and Bluetooth beacons and enable mobile interaction to be rapidly prototyped. The Microsoft Kinect depth sensor was released in 2010, and can be used to prototype detailed interactions in a limited area of a place. Bluetooth Low-Energy (BLE) beacons were announced in 2013, and can be used to prototype interactions that account for user presence in coarsely defined areas. Both sensors were new at the time of writing, and there were no alternatives to combine their sensing with a smartphone. Therefore, novel rapid prototyping tools were developed that integrate these basic sensing systems with a smartphone to allow applications to be built on top. The development of rapid-prototyping tools will enable a technology-driven investigation of the research questions. Positioning experiments were performed to evaluate the capabilities and limitations of both tools for detecting the smartphone user in places (RQ-4 of Section 3.1). Each tool was built with a sensor agnostic approach: It is intended that new sensing technologies can integrate with each framework as they become available, and interactions that are designed with these tools are independent from the sensing technique.

## 4.1 Rapid Prototyping with the Microsoft Kinect

Mobile interaction designers are encouraged to explore novel interactions that account for contextual information about a user, such as the position of the user in a room, in order to make interaction with devices more manageable. However, indoor positioning systems are not yet feasible for home environments, due to custom hardware requirements, and configuration requires expert knowledge and is costly in time and effort. Furthermore, embedding sensors in a room can be infeasible or disruptive, making it difficult for researchers to test interactions 'in the wild'.

The Microsoft Kinect provides a cheap and robust 3D sensor suitable for tracking humans in indoor settings. Using the Kinect to sense user behaviour is beneficial as it requires minimal equipment to sense many interactions and, as such, it is low-cost and causes little disruption to the working environment. The sensor was released in 2010 and is being used by millions of consumers. Since the hardware and development environments are widely available, any system developed to work with the Kinect can be shared with many other users. The Microsoft Kinect has been previously explored in the design of gesture systems with public displays, such as to mediate pointing with a customer in a retail information space [60]. It has been demonstrated that the limited sensing area of the depth sensor can be overcome by fusing the data of multiple Kinects and that the combined sensors of a mobile device can improve pointing recognition in a spatial environment [24].

Rapid prototyping is the process of simulating software design ideas quickly [146]. The Kinect can be used as a cheap indoor positioning system that is suitable for rapidly prototyping interactions with a mobile device in indoor environments [112]. Using the Kinect, it is possible to prototype a situated information space, such that the user experience can be abstracted from the sensing technology. With a single sensor, it is possible to design interactions that involve pointing, as has been demonstrated with IR sensing [3, 122], and positioning, as is possible with NFC tagging [157]. Table 4.1 suggests other examples of virtual sensors that would be possible with this system. An interesting advantage of connecting mobile devices to the framework is the potential to perform real-time user identification by synchronising device acceleration with hand position. The fine-grained control of the device may complement the coarse contextual data of the Kinect and allow for more interesting interactions to be explored.

| **Virtual Sensor** | **Kinect Data** |
|---|---|
| Proximity sensor | 3D position |
| Accelerometer | Second derivative of 3D position |
| Pose sensor | Skeleton tracking |
| Occupancy sensor | User detection |
| Motion sensor | Skeleton tracking |
| Light sensor | Camera |
| Sound meter | Microphone array |

Table 4.1: Examples of Virtual Sensors using Kinect data.

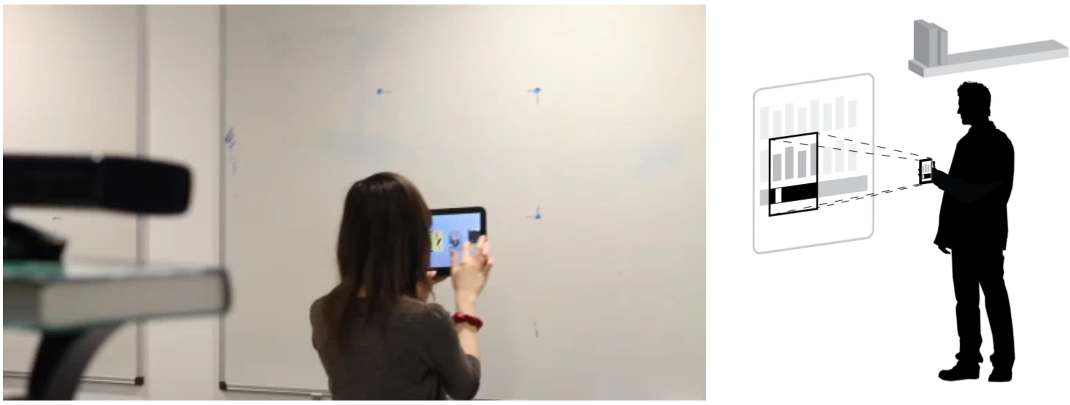


Figure 4.1: In the left figure, the Kinect is used to trigger a digital book collection on a peephole display. In the right figure, a peephole interface is situated at a static position beside a physical bookshelf.

### 4.1.1 Prototype Design: Kinect and Mobile Visualisation Tool

A tool is designed that gathers sensor data from the mobile device and Microsoft Kinect, as displayed in Figure 4.2. The tool has features to record and playback experimental conditions, so that results can be analysed in more detail. The application has two modules: one receives data from the Kinect, including the current position of the user, and the other responds to user interaction with the mobile device. The Kinect application manages a table of virtual sensors that are added using a mobile phone.

To illustrate the process of rapidly prototyping spatial interaction using the Kinect, a virtual bookshelf application is designed, as pictured in Figure 4.1. The bookshelf application allows a user to place a visual book library in a region of a room and explore this with a mobile phone acting as an augmented reality, peephole display [165]. In the final system, it is intended that the application will be triggered by a proximity sensor embedded into a physical bookshelf in the users' home. As it is not clear how the user should interact with bookshelf application around the physical bookshelf, or how sensitive the sensor will need to be, the idea is prototyped using a virtual proximity sensor. A mobile application can place a virtual sensor in the room by requesting to store an action at the current position of the user: to place the virtual bookshelf, the application requests to store the command 'bookshelf'. When the action 'bookshelf' is received, the smartphone knows that the user is in proximity of a virtual sensor and launches the virtual bookshelf application. This application could only be triggered when the user is in view of the Kinect. When the design stage is complete, the virtual sensor can be replaced or combined with a real sensor to extend the application possibilities and increase performance. The sensors available in a mobile device could also be considered. For example, coupling the position with a bearing-based direction [142] from a magnetometer could be used to determine when the user is facing the wall.

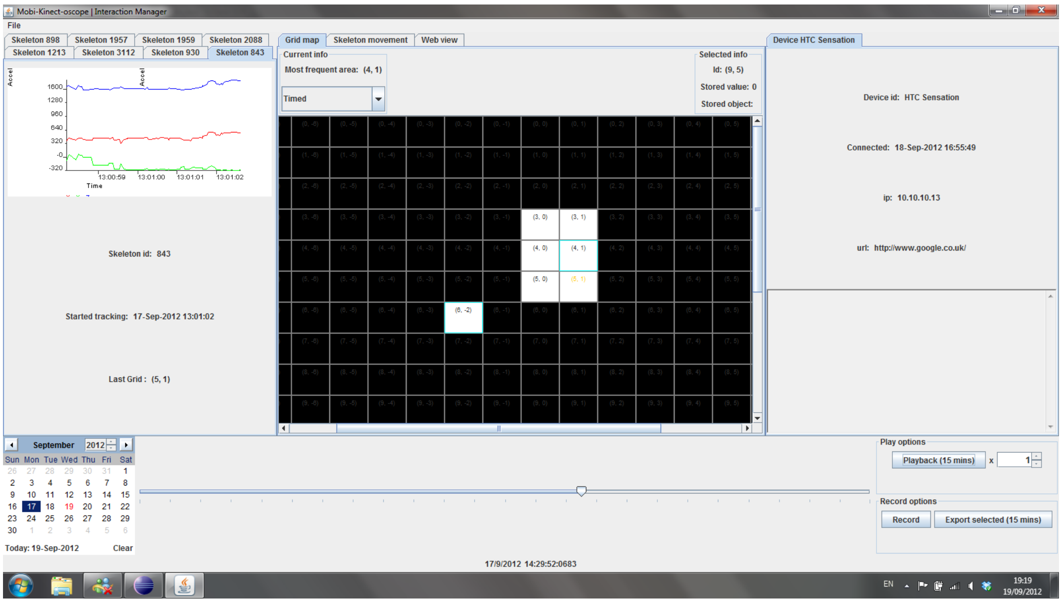



Figure 4.2: The tool for visualising and recording data from the Kinect and mobile device. The 2D grid locations hotspots that can be triggered when a Kinect Skeleton enters its (x, z)-position.

This simple message-based system for registering virtual sensors allows for many sensors and mobile applications to be registered using a single Kinect. The designer of the virtual bookshelf can quickly integrate spatial interaction into the mobile application and gain feedback on its use before committing to this part of the system.

### 4.1.2 Experiment: Evaluating Accuracy of Hand Positioning

The aim of this experiment was to determine the use of Kinect position data in practice by investigating the accuracy of users who were asked to select target positions with a mobile device. The experiment was run with 3 participants, with mean age of 35, mean height of 172cm and 1 was female. 120 sample positions were generated for 2 fixed points.

#### Method

The experiment setup is shown in Figure 4.3 and was consistent over all participants. The following equipment was used in the experiment: a Microsoft Kinect sensor, an Android mobile phone, a laptop installed with the Primesense OpenNI software and a wireless network. The laptop was attached to the Kinect via USB and communicated with the mobile device over a wireless network. The Kinect sensor was an approximate distance of 3 meters from the whiteboard and was placed in a location such that its view was of the testing area. The whiteboard was marked with two points: Point 1 located at the intersection of the optical

Figure 4.3: Experiment setup. The participant is pointing the mobile device in front of Point 1 in the forward facing condition and is being tracked by the Kinect.

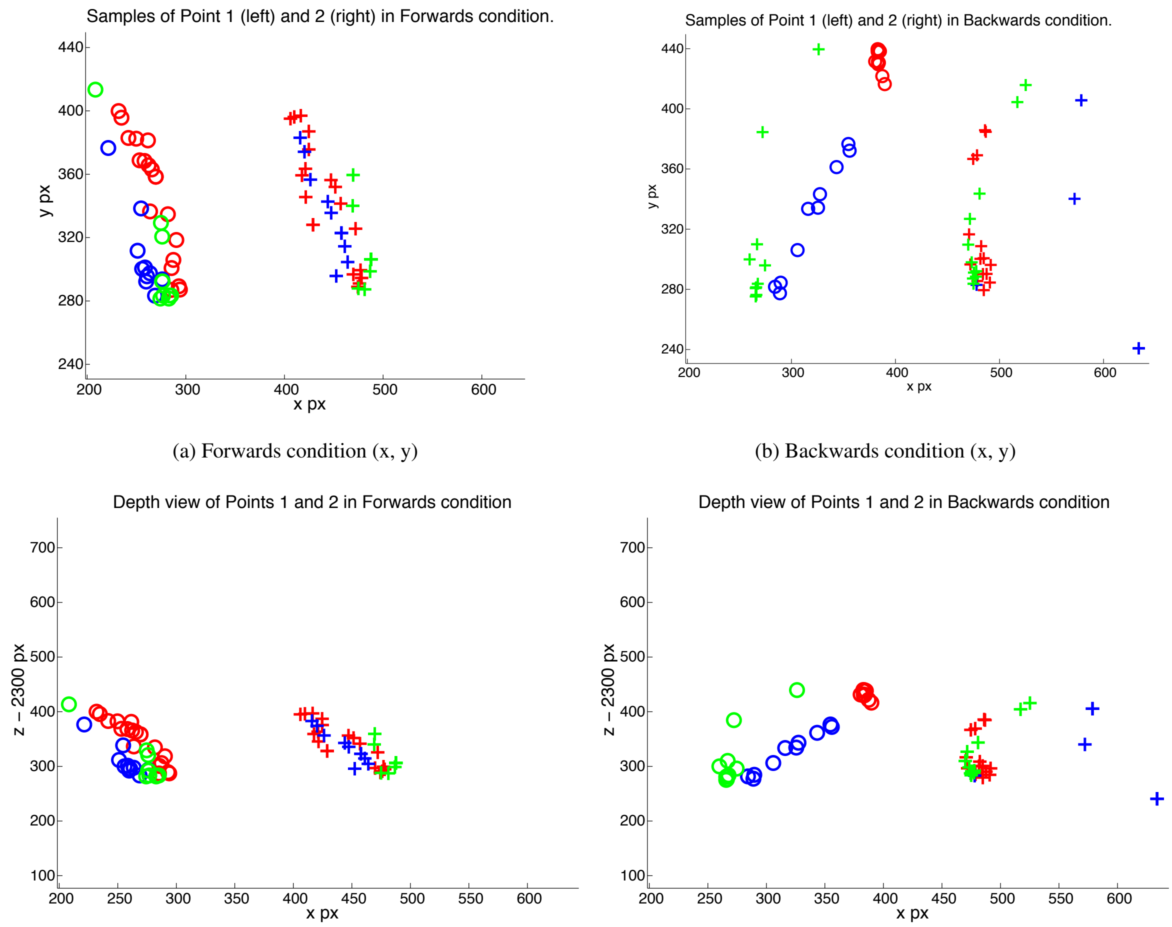
axis of the Kinect and the whiteboard, and Point 2 measured 1 meter left of Point 1. The distance of the Kinect scaled the position data to approximately 1px:2cm. To start the tracking process, participants were required to perform the OpenNI PSI pose, a stance where the arms are held at a $90^\circ$ towards the Kinect, in order to calibrate the skeleton data.

#### Task

Participants were asked to reach the mobile device in front of a point on the whiteboard and press a button on the device; when the button was pressed, the position of the right hand was recorded from the Kinect skeleton data. Participants received vibrotactile feedback to confirm this selection. The experiment investigated two conditions: facing forwards towards the Kinect and facing towards the whiteboard. Using the right hand forced movement direction to change and this is noticeable in the results. Each task was repeated 10 times consecutively for each condition and both points.

### 4.1.3 Results

The graphs in Figure 4.4 display all point samples and show the separation between Point 1 and 2, indicating that these two points could be classified uniquely. These graphs also reveal




(a) Forwards condition (x, y)

(b) Backwards condition (x, y)

(c) Forwards condition (x, z)

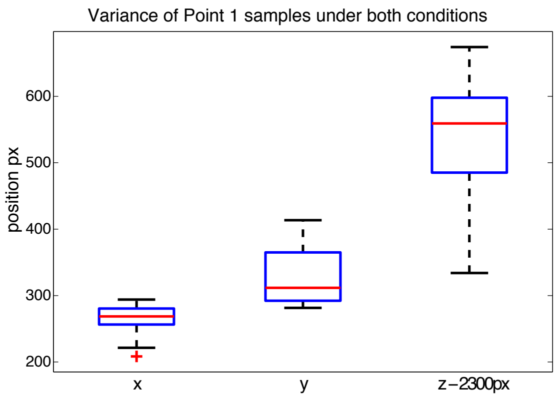
(d) Backwards condition (x, z)

Figure 4.4: Each graph shows sample positions of Point 1 and Point 2. Each point is represented consistently by symbol and participants are represented by colour. Positions ranged between (0,0,0) and (640, 480, 3200) pixels. The graphs have been scaled to focus on the results.

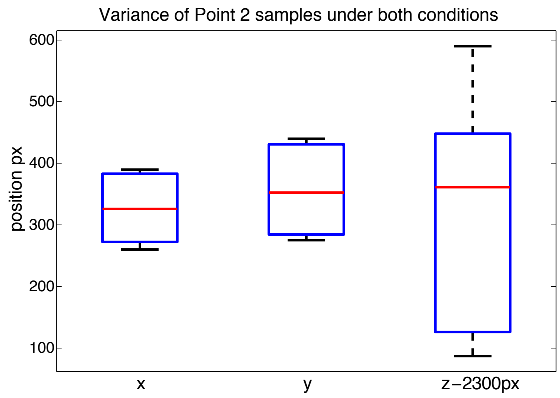
noticeable differences between the forward and backward facing conditions: the direction of the hand movement caused the samples to be characteristically skewed. This effect was due to a communication delay between the phone and the laptop and has been improved with a more reliable communication protocol; this could be analysed further by synchronising the sampled positions with the accelerometer movement of the mobile phone. The bottom graphs in Figure 4.4 illustrate hand movement in the z-axis: there were no significant effects due to the position of Point 2, which was angled away from the Kinect sensor.

The variation of the 3D sample positions is shown by the boxplots in Figure 4.5. Samples were measured to be within a maximum distance of 13cm, suggesting that a range would need be set in order to uniquely identify a position. This range would restrict the use of position data to applications requiring a point separation greater than this measurement.

Participants were asked to comment on the idea of a virtual bookshelf prototype and promis-




(a) Point 1




(b) Point 2

Figure 4.5: Variance between all samples of Point 1 and Point 2 under both forwards and backwards conditions.

ing comments were received, though the idea of the application itself did not appeal to all. Some participants were happy to access their content through their standard touch interactions and did not see the spatial interaction as an advantage. However, the potential of coupling the Kinect with a mobile device was highly inspiring and participants were readily able to imagine their own ideas of applications: One participant travelled often and wanted the application as a method of home security using information on the number of users detected. Another participant worked late and wanted to use the system as a social tool by revealing secret messages to their partner in the home. In a business perspective, a participant was interested in real-time user identification and how it could allow both shops and visitors manage their visits automatically and build on the 'check-in' facility of Foursquare, a location-based social network [96].

### 4.1.4 Discussion

It was demonstrated that the Kinect can be used to simulate a proximity sensor, and the variability of sampled positions was tested when users targeted a point on the wall. The results indicate that the system can create compelling augmented reality systems, where the combination of mobile phone and Kinect are used to emulate a range of virtual sensors. With the framework, users can turn any surface or object in a room into digitally augmented ones, where the physical object or area acts as a mnemonic device for the content or service, as illustrated in the example of a virtual bookshelf application. The ease of placing virtual sensors allows users to immediately customise a new room, such as a hotel or meeting room, to have augmented content that is accessible from any mobile device. Simulating with virtual sensors allows for many of these interaction ideas to be explored early in the design stage.

The limitations on this approach are common to most vision-based techniques. The static sensing area of the Kinect forces interactions to be bound to a fixed location and there are limitations on the range of depth sensing. Additionally, there can be problems with obstructions in the environment. Combining the inertial sensors of a mobile device would allow us to overcome some of these issues.

One way in which the framework could be extended is to allow interactions with multiple users and multiple devices, which allows much more complex and interesting interactions to be rapidly prototyped. Correlating the acceleration of each device with the motion of the Virskeletons sensed by the Kinect would allow immediate identification of users. Virtual sensors other than position-based sensors could also be implemented. A lightweight framework for position sensing will encourage the design of mobile interaction applications based on the low cost standard components involved and the sharing of novel interaction ideas.

## 4.2 Detecting Places with Bluetooth Beacons

Bluetooth is a pervasive technology that can be found in many devices including laptops, smartphones and wearable technologies. Though traditionally Bluetooth was designed for short-range communication, it has been explored for indoor-positioning. For several reasons traditional Bluetooth has been considered ill-suited for accurate location sensing [81, 103, 91]. Hardware that was designed for Bluetooth communication was inflexible to configure a positioning system, RSSI values were not standardised between hardware and were not comparable, and the limited number of beacons were sparsely deployed in fixed positions, and could have dynamic availability: for example, in the instance of a desktop PC that is turned off at the end of each day. Furthermore, wireless signals interfere with the Bluetooth signal.

It is possible to deploy a system with traditional Bluetooth beacons that can detect a 2 - 3m room, by averaging the Received Signal Strength Indicator (RSSI) [6]. To achieve higher accuracy indoor positioning, manual finger-printing can be performed [92, 129], involving sampling a database of signal strengths at discrete positions in a room. Alternatively, specialised hardware can be installed, such as radio transmitter in the ceiling of a room [74]. However, such solutions are costly to set up, and there is a requirement to make indoor positioning systems quick and simple to deploy [104, 76, 22] so that real-world applications that use this context can be explored.

The complicated setup of indoor positioning systems has limited the opportunities to conduct user studies in personal environments. With the release of Low-Energy Bluetooth (LE) beacons in 2014, there is a potential to design a rapid-prototyping tool that detects the smartphone user in personal places, and that is quick and easy to set up. An approach to detecting

| Design Stage | Design Factors |
|---|---|
| Configuration | Scan frequency and duration, beacon TX Power and interval, distance calculation. |
| Deployment | Place detection accuracy, signal interference, position and quantity of beacons. |
| User Experience | Perceived accuracy, beacon and device battery drain, set up cost. |

Table 4.2: Factors to consider when designing for place detection.

places is presented that could be easily deployed with little set-up costs or disruption to the working environment. This would be appropriate for rapid-prototyping interactions that use the context of personal places. The proposed rapid-prototyping approach involves tracking a small number of personal places on a smartphone, by maintaining a short list of known beacons. Using information about where smartphone interactions are most likely to occur and attenuation caused by structures of the environment, beacons are positioned in a way that is likely to minimise signal overlap. The experiments below demonstrate the approach and the tools involved. It is intended that the interactions designed with this approach can be used with more reliable sensing algorithms before being deployed in the wild.

### Design Factors

There are various factors to consider when using Bluetooth to perform room-level positioning, including the configuration, deployment and user experience, as summarised in Table 4.2. Place trackers scan in the background of the mobile device over long periods of time. This will have implications for battery power of the mobile device. Therefore, it is important to consider the frequency and duration of Bluetooth scans when implementing the place tracker. Similarly, Bluetooth LE beacons transmit advertisements at a certain frequency and power, will affect the battery life of each beacon. The transmission power (TX power) of a beacon has an impact on the size of place that a single beacon can be used to detect. A lower TX power will have a shorter range, which will consume less power and will interfere less with neighbouring places. The range of the Bluetooth beacon will have implications for the accuracy of the place tracker.

In personal places, environment conditions are dynamic, with other wireless signals, people walking in front of beacons, and the opening and closing of doors. Dynamic conditions cause interference to the Bluetooth signal, and will impact the reliability of the detection algorithm when the place detection system is deployed.

The user experience of the positioning system is also important to consider as it will impact the acceptance of the system. When the system is to be installed in personal places, such as rooms in a household, the user experience will include how many beacons are required, where they should be placed and the involvement of the user in setting up the system.

### Traditional Bluetooth beacons

The JAKE sensor displayed in Figure 4.6 (a) contains a Bluetooth transmitter, among other sensors. A benefit of using the JAKE as a beacon compared to a notebook [48] or headset [29] is it is very small and can be easily concealed, making it more suitable for long-term deployment in a home setting. Additionally, JAKE beacons do not require a desktop PC for power, which are often powered intermittently [8]. A drawback of the JAKE beacons is their short battery life, and so they need to remain plugged in to a power source if they are to be used for several hours.

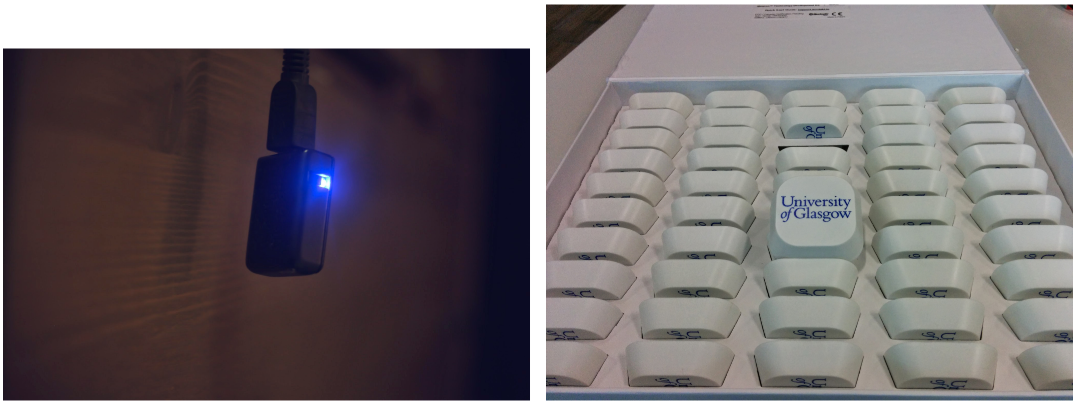
### Low-Energy Bluetooth Beacons

In June 2013, the Apple iBeacon protocol[1] was announced, and Low-Energy Bluetooth (LE) beacons soon became popular for marking positions indoors. A benefit of the low-energy beacons over traditional Bluetooth is that the transmission power of the beacons can be configured, and their communication range can be lower than is possible with traditional Bluetooth beacons. Additionally, Bluetooth LE beacons can last several months or years on a single battery, and therefore can be self-contained units that are not constrained by their power source. Kontakt.io were one of the first manufacturers to adopt the iBeacon standard, and an example of a Kontakt.io beacon is displayed in Figure 4.6 (b). Applications with Bluetooth LE beacons have been explored for commerce. For example, a supermarket might put a Bluetooth beacon next to a special promotion, and nearby shoppers with the appropriate app installed can be notified just-in-time.

## 4.2.1 Prototype Design: My Places

My Places is an Android mobile app that is designed to assign logical labels to a set of Bluetooth beacons, and to determine which is closest to the smartphone user. Bluetooth supports the naming of devices [144]. However, as these names are publicly discoverable, and there could be privacy implications in revealing the relationship of a user to a place. Additionally, physical places can have different meanings to different people, and so there may be disagreement over which name to choose if others share a place. Services like Foursquare[2] allow customers to name a place when they check-in to a GPS location. Similar to this approach, My Places allows the same beacon to be named differently by each individual, by

[1] https://developer.apple.com/ibeacon
[2] http://foursquare.com




(a) JAKE Bluetooth beacons.

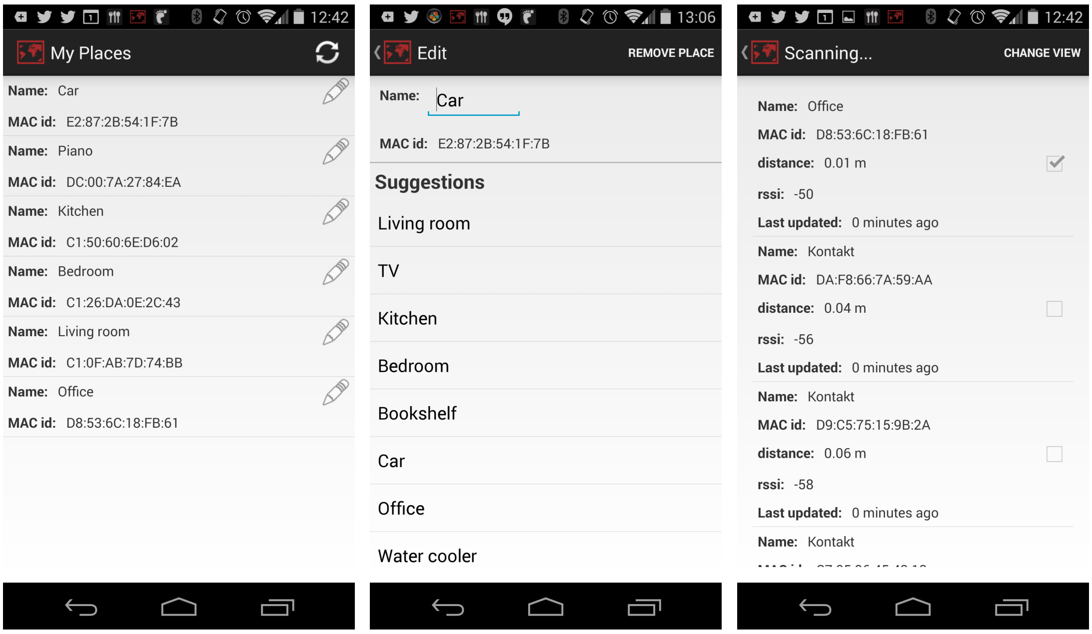
(b) Kontact.io Low Energy Bluetooth beacons.

Figure 4.6: Bluetooth beacons.

storing this relationship in the app. Furthermore, an individual can label multiple beacons with the same name, allowing a larger space to be detected seamlessly with multiple beacons. My Places also makes it simple for the user to change the label of a beacon, if a beacon should be moved to a different place.

An example of beacons that have been assigned labels with the My Places app can be seen in Figure 4.7 (a). Nearby beacons can be found in the scan menu, as displayed in Figure 4.7 (c), with the results ordered by distance. Clicking on a result adds the beacon to a database of known beacons, and presents a menu to assign a label to the beacon, or remove it from the list, shown in Figure 4.7 (b).

My Places can share its scan results and current place estimate with other apps. A separate evaluation tool was designed to integrate with My Places, as displayed in Figure 4.8 (a). This application receives the Bluetooth scan results from My Places, and filters for the experiment beacons prefixed with ‘beacon exp’. When the checkbox is selected, data is logged to a .csv file. In addition to the scan results and the current place detected by My Places, the log includes the latest readings from the accelerometer and orientation sensors and the battery level of the device. An optional message can be appended to annotate the sensor data with details about the position of device in relation to the beacons. The scans can be limited by time or by the number of samples, and will automatically stop once enough samples have been recorded. A vibration will alert the experimenter when the scan is complete. A separate view plots the readings from each beacon over time, and can be used to visualise any signal overlap, as displayed in Figure 4.8 (b). The evaluation app also allows the experimenter to configure the interval and TX Power of the Kontakt.io beacons.



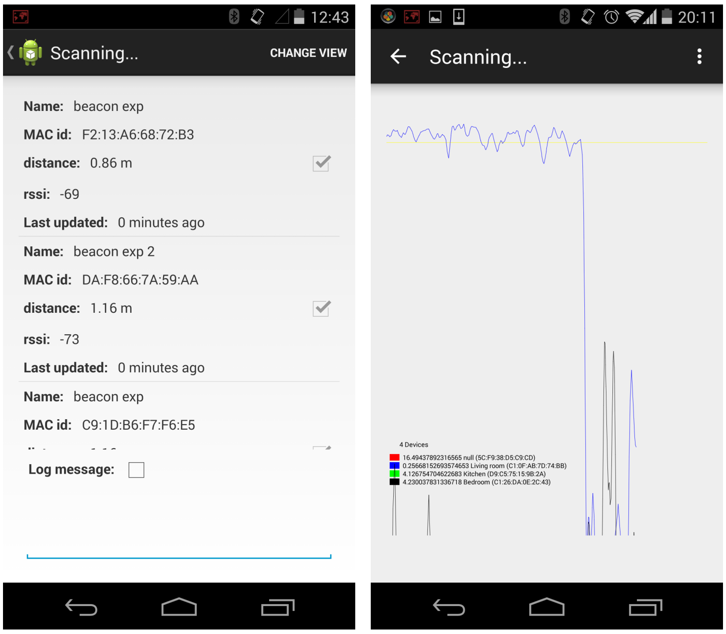

(a) List of beacons added to My Places. (b) Edit menu. (c) Scan menu.

Figure 4.7: Beacons could be added to the list (a) by selecting them in the scan menu , where beacons are ordered by distance. The edit menu (b) is opened by clicking on an entry that has been added or discovered during a scan. Suggestions of place labels could be clicked to name beacons quickly, or a custom name can be entered in the text box.

### 4.2.2 Experiment: Evaluating Place Detection Accuracy

The structure of each physical environment poses different challenges for detecting personal places. If a place is neighboured by others, such as two rooms in an apartment, then the smartphone may be falsely detected in the neighbouring place if it is positioned near its boundaries. A hallway might not be considered a personal place, and as such one might not choose to detect a hallway. However, without a beacon to mark the hallway, an adjacent place might be detected if the device is used as it passes through. The detection of places may be simplified in a car or an office if there are no other personal places to detect nearby. However, in such places, it should be considered when it counts to be inside or outside.

The proposed approach of detecting places considers the positions in a room where the user likely to interact with a smartphone. For example, a user might launch apps on the smartphone while sitting on a couch in a living room, compared to standing directly in front of a television. Similarly, a kitchen table might be a more popular landmark for interacting with a smartphone, compared to the kitchen sink. These insights are used to rapidly prototype place detection. The challenges of room-level positioning with Bluetooth beacons in a home setting are highlighted in the experiment.




(a) List view.

(b) The RSSI signal can be displayed as a plot.

Figure 4.8: The evaluation app can log sensor data alongside data received from My Places.

### Research method

To prototype interactive applications, the place tracker should scan frequently enough to detect advertisements from nearby beacons, and often enough to detect a change in place in real-time. To theorise about how quickly a human can move between places, the walking features of an average human can be considered. [13] found the mean preferred walking speed to be 3.90 km/h (1.08 m/s), and 2.97 km/h (0.83 m/s) while interacting with a touch-screen. If the average stride length of a human is less than 1m, it could be reasoned that a human could exit one place and enter another in approximately 1s. It might then be desirable to select 1s as the maximum interval between scans to immediately detect a user entering a place. However, one should note that there is a trade-off to be made between scan frequency and battery power [76]. Additionally, RSSI signals are noisy, and so an average of several samples should be taken, which has an impact on how quickly one can detect a change in place. For the JAKE beacons, several scans is often required to discover all nearby devices [144].

Figure 4.9 presents a configuration that scans for RSSI signals for a duration of 0.5s, and stops for an interval of 1s. If a user moves into a new place when a scan is scheduled, then it should be detected immediately; changes to place that occur when the scan has stopped will be discovered in the next scan, assuming that the scan interval is not long enough for the user to leave. Two configurations of beacons are shown: One beacon advertises at a rate less than the scan duration, and is detected during every scan. Another beacon advertises at a rate equal to the scan duration, and is missed during every second scan. It should be sufficient

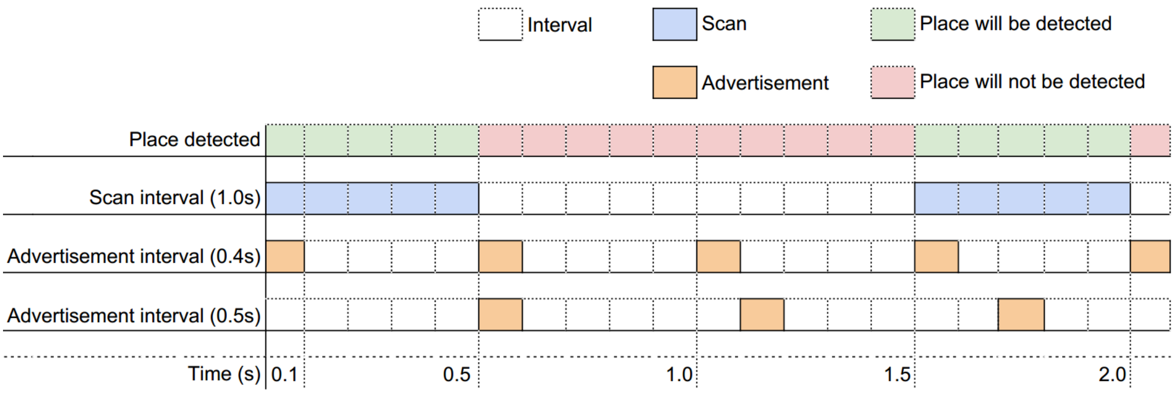



Figure 4.9: The beacon advertisement will be detected during every scan if its interval is less than the scan duration. Changes to place that occur after the scan has stopped will be discovered during the next scan.

to select a scan duration greater than beacon interval to detect advertisements during every scan.

### Configuration: Bluetooth LE Beacon

Though the frequency and TX power of LE advertisements can be controlled, the default advertisement values for the Kontakt.io beacons are used in the experiment: -12dBm and 350 ms. Kontakt.io describe these settings as the best compromise between battery life and user experience. My Places scans for LE beacons for a duration of 0.5s followed by an interval of 1s, using the Android startLeScan()[3] function. At the end of each LE scan, a moving average of the RSSI signal is computed, and the current place estimate is determined as $\min_{\forall B \in S} \left(\overline{\mathrm{B}_{distance}}\right)$, where $S$ is the set of beacons that advertisements were received from in the last 5s, and $\overline{\mathrm{B}_{distance}}$ is the mean distance calculated over a sliding window of 30s.

Apple iOS devices calculate the distance estimate to an iBeacon internally. On Android, this calculation must be performed manually for every advertisement packet received. High accuracy distance calculations can require additional infrustructure and calibration to account for variability in mobile devices and the user environment. To perform room-level positioning, only relative distance between devices is required, and so distance calculations that require timely calibration for the user are not used. Instead, a simple power function was found to be suitable, that is based on the formula defined in the Android iBeacon service by Radius Networks[4]:

$$\mathrm{B}_{distance} = \left(\frac{\mathrm{B}_{RSSI}}{\mathrm{B}_{meter}}\right)^{10} \tag{4.1}$$

where $\mathrm{B}_{distance}$ is the distance estimate (m), $\mathrm{B}_{RSSI}$ is the RSSI (dBm) and $\mathrm{B}_{meter}$ is the RSSI

[3] `http://developer.android.com/reference/android/bluetooth/BluetoothAdapter.html#startLeScan(android.bluetooth.BluetoothAdapter.LeScanCallback)`

[4] `http://developer.radiusnetworks.com/2014/12/04/fundamentals-of-beacon-ranging.html`

(dBm) measured at 1 meter for a given TX power. In the configuration for the experiment, the RSSI at 1 meter was estimated to be -68 dBm. This distance estimate provides a more human-friendly understanding of distance to the beacons compared to RSSI. A robust application will use a more advanced approach to estimate distance to the beacon.

**Configuration: Traditional Bluetooth Beacon**

Unlike the LE beacons, a full Bluetooth inquiry must be performed to detect JAKE beacons. The Android startDiscovery()[5] function performs a Bluetooth discovery for a duration of approximately 12s, followed by an interval of 1s. At the end of each discovery, which JAKE beacon is closest is decided with: $\min\limits_{\forall B \in D}(|\mathrm{B}_{RSSI}|)$, where $D$ is the set of beacons that were received during a Bluetooth discovery, and $|\mathrm{B}_{RSSI}|$ is the absolute RSSI value received from a beacon B.

### Research design

Three experiments were performed in the author's apartment, displayed in Figure 4.10. The apartment was contained on the ground floor of a three storey detached block that housed 6 units. The walls separating the rooms were modern timber stud design plasterboard lined on both faces. Three places were selected for the experiment: Lounge (4 x 3.5m), Kitchen (3 x 2.5m), and Bedroom (3.5 x 3m).

**Deployment**

One beacon marks each place, and their positions are highlighted as coloured squares in Figure 4.10. The positions were chosen to be somewhere that fit the home layout, and were far enough apart to avoid overlap between neighbouring beacons. The beacon in the Kitchen ($K$) was positioned on top of a microwave in the kitchen, away from the sink and cooking facilities. The beacon in the Lounge ($L$) was placed by the television opposite the couch. The beacon in the Bedroom ($B$) was fastened with Blu-tac against a chest of drawers next to the bed.

**Structural Interference**

To compare the signals of beacons in neighbouring rooms, a record was made of the distance and RSSI received by a smartphone that was positioned at measured distances between two rooms in an apartment. Figure 4.11 (a) shows the set up of these experiments, and the beacons in each room. 6 positions were chosen at 1m apart, with 3 in each room. Each position was measured from the beacon in the Lounge. The My Places evaluation app was installed on a Nexus 5 running Android 5.0 and was used to collect samples for 3 minutes at each distance, with both the LE beacons and the JAKE beacons compared.

[5]`http://developer.android.com/reference/android/bluetooth/BluetoothAdapter.html#startDiscovery()`


(a) Kitchen (b) Lounge

(c) LE beacon positions (d) Bedroom

Figure 4.10: Experiment rooms with three LE beacons: (a) Kitchen (K, black), (b) Lounge (L, green), (d) Bedroom (B, red)

### Landmarks

Measurements at the most likely landmark positions in three rooms, and in a passing place, are also performed. The set up of this experiment is displayed in Figure 4.12. In the Kitchen, the smartphone was placed on a surface facing away from beacon K. In the Lounge, the smartphone was placed on a small table facing towards beacon L. In the Bedroom, the smartphone was placed on a bed, facing perpendicular to beacon B. In the hallway, the smartphone was placed on a table, facing towards the Lounge. The My Places evaluation app was used to collect 250 samples at each position.

### Walking

To consider the accuracy of the place detection, sensor readings were gathered while walking slowly around the edge of the available space in each place in the apartment. The set up is displayed in Figure 4.13 (a). A slow walking speed was chosen as it is most appropriate for a household, and as it would increase the risk of interference from neighbouring beacons. We used the My Places evaluation app to annotate the scans with ground truth about each place.

The accuracy was calculated to be the number of times that My Places correctly detects the current place. 250 samples were recorded in the three places.

**Advertisements**

The LE beacons transmit advertisements every 0.35s, and so each beacon will transmit $\lfloor \frac{3*60}{0.35} \rfloor = 514$ advertisements in a 3 minute period. My Places performs LE scans every 1.5s for a duration of 0.5s, and in 3 minutes the evaluation app will record the results from 120 scans. Therefore, it was expected to receive $120 \leq n < 180$ advertisements per beacon, i.e. at least 23% of LE beacon advertisements. With the JAKE beacons, a 12s Bluetooth discovery was performed every 13s, and it was expected that each JAKE beacon would transmit at least one advertisement during each discovery. Therefore, at least $\lfloor \frac{3*60}{13} \rfloor = 13$ responses from JAKE beacons were expected in a 3 minute period.

### 4.2.3 Results

The measurements taken with LE beacons are presented, starting with the results at measured distances, at stationary positions and when walking. The results comparing the LE beacons to the JAKE beacons follow.

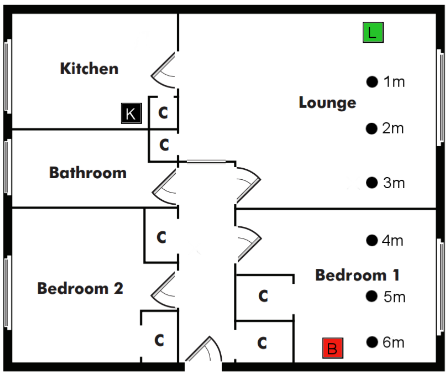
#### Bluetooth LE Beacons

**Structural Interference**

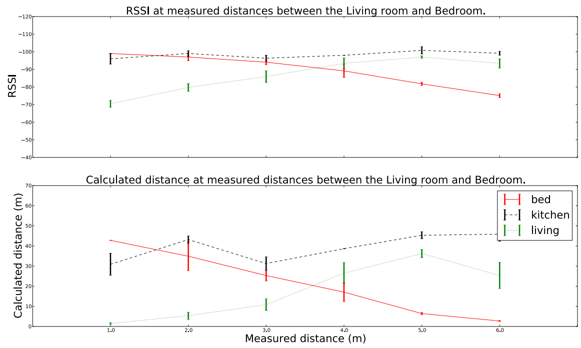
Figure 4.11 shows the RSSI and calculated distance measurements recorded at increasing measured distances in the Lounge and the Bedroom. The signal from $L_{LE}$ shows a clear rise in $|RSSI|$ and calculated distance as the smartphone moves towards the Bedroom. Similarly, $B_{LE}$ shows a clear decrease in $|RSSI|$ and calculated distance. A wall separates the Lounge and Bedroom at 3.5m from $L_{LE}$, and the $L_{LE}$ and $B_{LE}$ signals cross over at 4m where the smartphone was moved into the other room. There is also an increase in variance at the positions at either side of the wall. This suggests that the attenuation between rooms is enough to shield the signals from each beacon.

**Landmarks**

Figure 4.12 shows the RSSI and calculated distance measurements recorded at stationary landmarks in four rooms. $B_{LE}$ is not detected in the Kitchen, and $K_{LE}$ is not detected in the Bedroom or the Hall. The nearest beacon inside each room is detected as being less than 12m away. Neighbouring beacons are detected as being greater than 20m away, and are detected in fewer scans than the beacon in the same room as the smartphone. Similarly, in the hall, beacons are detected to be further than 20m away. In this stationary condition, the RSSI and




(a) Set up




(b) RSSI (Top) and Calculated distance (Bottom)

| **Distance measured from $L_{LE}$** | $L_{LE}$ | $B_{LE}$ | $K_{LE}$ |
|---|---|---|---|
| **1m - Lounge** | -70 (var=3.8) | -99 (var=0.0) | -96 (var=8.6) |
| **2m** | -80 (var=4.8) | -97 (var=4.8) | -99 (var=2.5) |
| **3m** | -86 (var=11.3) | -94 (var=2.0) | -96 (var=2.8) |
| **4m - Bedroom** | -93 (var=9.4) | -89 (var=13.9) | -98 (var=0.0) |
| **5m** | -97 (var=0.4) | -82 (var=0.8) | -101 (var=4.2) |
| **6m** | -93 (var=7.1) | -75 (var=1.2) | -99 (var=1.5) |

(c) Mean RSSI and variance of three beacons, at distances measured from $L_{LE}$ in the Lounge.

| **Distance measured from $L_{LE}$** | $L_{LE}$ | $B_{LE}$ | $K_{LE}$ | $Total$ |
|---|---|---|---|---|
| **1m - Lounge** | 128 | 1 | 42 | 171 |
| **2m** | 137 | 65 | 33 | 235 |
| **3m** | 114 | 85 | 47 | 246 |
| **4m - Bedroom** | 39 | 40 | 1 | 80 |
| **5m** | 5 | 132 | 5 | 142 |
| **6m** | 74 | 139 | 6 | 219 |

(d) The number of advertisements received from each LE beacon at measured distances in a 3 minute period.

Figure 4.11: RSSI and calculated distance measured at increasing distances from $L_{LE}$: Lounge ($L_{LE}$, green dotted), Bedroom ($B_{LE}$, red solid) and Kitchen ($K_{LE}$, black dashed).

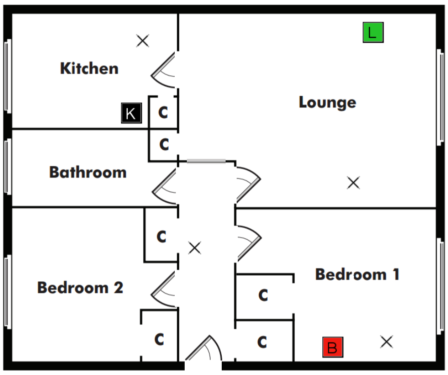
calculated distance measurements correctly indicate the beacon in each room as being the nearest.

### Walking

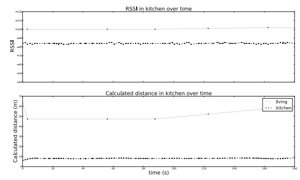
Figure 4.13 displays the signals received while walking slowly in a circle around the edge of three rooms. The walking path is displayed as an orange line. $B_{LE}$ was not detected in the Kitchen. In all rooms, there are instances where the RSSI signals overlap. The calculated distance measurement smoothes out the RSSI signal, and indicates that the beacon in each room is detected as the nearest.

### Advertisements

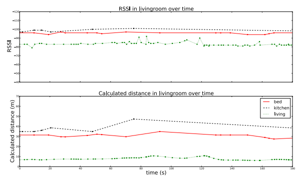
Table 4.11 (d) summarises the number of LE advertisements received at each measured dis-



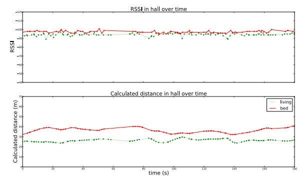

(a) Set up



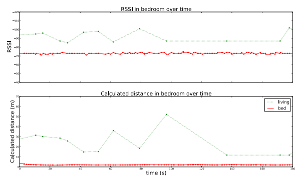
(b) Kitchen



(c) Lounge



(d) Hall



(e) Bedroom

| Distance (m) | Kitchen | Lounge | Bedroom | Hallway |
|---|---|---|---|---|
| $K_{LE}$ | 8.07 (SD=0.41) | 34.72 (SD=5.39) | - | - |
| $L_{LE}$ | 48.8 (SD=7.51) | 8.73 (SD=3.14) | 24.19 (SD=13.73) | 25.89 (SD=2.57) |
| $B_{LE}$ | - | 30.70 (SD=2.45) | 2.05 (SD=0.18) | 36.04 (SD=2.83) |

(f) Average calculated distance (m) for three LE beacons

Figure 4.12: Average calculated distance (m) for three LE beacons, $L_{LE}$ (green dotted), $B_{LE}$ (red solid) and $K_{LE}$ (black dashed), at stationary positions marked X in four rooms: (b) Kitchen, (c) Lounge, (d) Hallway and (e) Bedroom.

tance. A maximum of 139 advertisements were received, which is within the expectations for a 3 minute scan. However, many advertisements are not discovered during every scan, particularly as the device moves further away from the beacon: 82.8 (SD=48.3) from $L_{LE}$, 77.0 (SD=48.7) from $B_{LE}$, and 22.3 (SD=18.8) from $K_{LE}$.

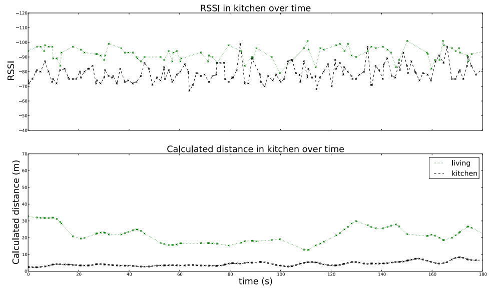
### Traditional Bluetooth Beacons

#### Structural Interference

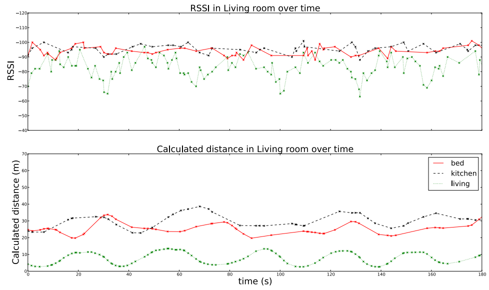
Figure 4.14 displays the average RSSI signal at increasing distances from $L_{JAKE}$. The relationship between RSSI and measured distance is less apparent. However, the signals cross over at 3m where the smartphone is approaching the Bedroom. Greater measures, such as shielding [29], will be required to reduce the range of the JAKE beacons to avoid overlap between the $L_{JAKE}$ and $B_{JAKE}$.

#### Advertisements

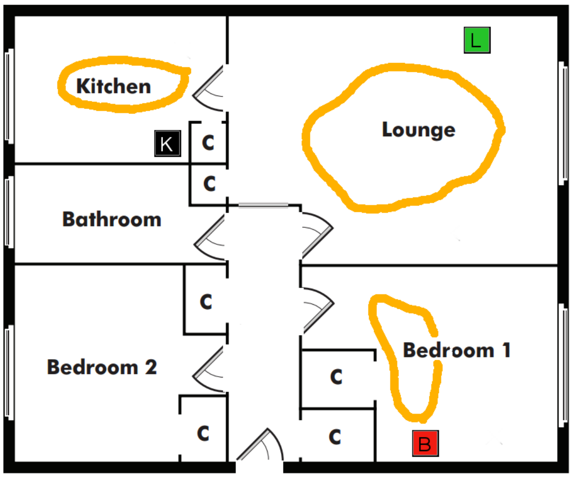
Table 4.14 (d) summarises the number of JAKE responses received at each distance. A total of 22 (SD=1.8) responses were received at each position: 8 (SD=1.3) from $L_{JAKE}$ and $B_{JAKE}$, and 6 (SD=1.6) from $K_{JAKE}$. A maximum of 10 responses were received in a 3 minute discovery, which is lower than the 13 that was expected. This may be due to



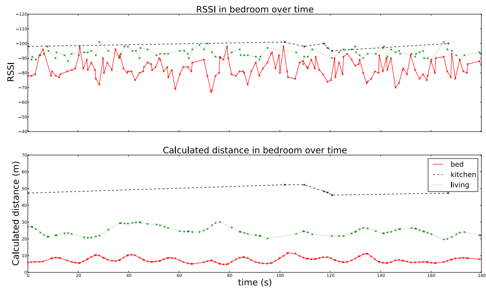

(a) Kitchen




(b) Lounge




(c) Set up




(d) Bedroom

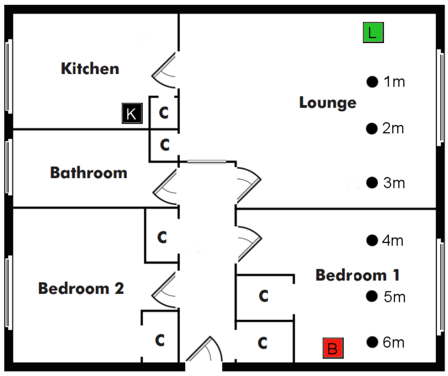
Figure 4.13: Walking in a circle in three rooms: (a) Kitchen, (b) Lounge and (d) Bedroom. Three LE beacons are detected in the Lounge (b) and Bedroom (d) : $K_{LE}$ (black dashed), $L_{LE}$ (green dotted) and $B_{LE}$(red solid). The walking path is displayed as an orange line.

interference, and that several scans may be required to discover all nearby devices [144].

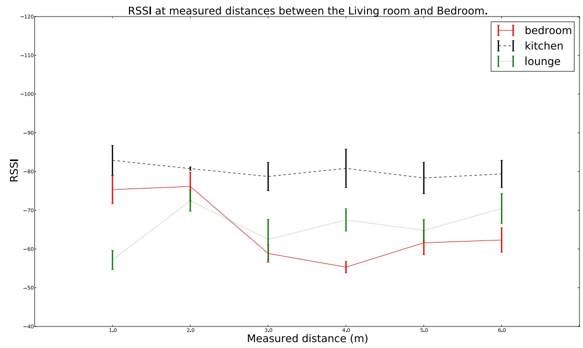
### 4.2.4 Discussion

A rapid prototyping approach to detecting places with Bluetooth beacons was demonstrated, and data was gathered in a home setting using the My Places tool, with three Bluetooth beacons positioned in a Kitchen, Lounge and Bedroom. This approach can be used to detect a small number of personal places, by maintaining a short list of known beacons. The position of beacons were chosen by using information about where smartphone interactions are most likely to occur and attenuation caused by the structures of the environment to minimise signal overlap. This approach could be easily deployed with little set-up costs or disruption to the working environment, and is appropriate for rapid prototyping interactions that use the context of personal places. It is intended that interactions designed with this tool can transfer to a more accurate positioning system.

Data was collected with an evaluation tool in a home setting, and the challenges of detecting




(a) Set up




(b) RSSI (Top) and Calculated distance (Bottom)

| Distance measured from $L_{JAKE}$ | $L_{JAKE}$ | $B_{JAKE}$ | $K_{JAKE}$ |
|---|---|---|---|
| **1m - Lounge** | -58 (var=5.4) | -77 (var=15.2) | -84 (var=13.0) |
| **2m** | -72 (var=4.4) | -75 (var=10.4) | -82 (var=4.7) |
| **3m** | -63 (var=31.4) | -59 (var=5.3) | -80 (var=17.7) |
| **4m - Bedroom** | -69 (var=14.8) | -56 (var=1.6) | -81 (var=21.3) |
| **5m** | -69 (var=12.5) | -64 (var=79.3) | -80 (var=16.3) |
| **6m** | -72 (var=17.4) | -63 (var=7.4) | -79 (var=10.9) |

(c) Mean RSSI and variance of three beacons, at distances measured from $L_{JAKE}$ in the Lounge.

| Distance measured from $L_{JAKE}$ | $L_{JAKE}$ | $B_{JAKE}$ | $K_{JAKE}$ | $Total$ |
|---|---|---|---|---|
| **1m - Lounge** | 10 | 10 | 5 | 25 |
| **2m** | 10 | 10 | 3 | 23 |
| **3m** | 7 | 7 | 7 | 21 |
| **4m - Bedroom** | 7 | 7 | 7 | 21 |
| **5m** | 7 | 7 | 6 | 20 |
| **6m** | 8 | 8 | 8 | 24 |

(d) The number of responses received from each JAKE beacon at measured distances in a 3 minute period.

Figure 4.14: RSSI and calculated distance for JAKE beacons in the Lounge ($L_{JAKE}$, green dotted), Bedroom ($B_{JAKE}$, red solid) and Kitchen ($K_{JAKE}$, black dashed).

places with Bluetooth was highlighted. The RSSI signals of LE beacons overlapped while walking at a slow pace. However, averaging over samples smoothed the RSSI value and avoided false positives. It was found that the RSSI signal from LE beacons can be accurate enough to perform room-level positioning indoors under these constraints. In comparison, the RSSI signals received from traditional Bluetooth beacons were more noisy, and the RSSI signals overlapped even while the device was stationary.

The release of LE beacons has increased the possibilities for detecting indoor locations. The ability to control the frequency and power of advertisement packets increases the flexibility for interaction designers. The long battery life makes it possible to position the beacon anywhere and is appropriate for in-the-wild studies, including places with an intermittent power supply, such as a car.

A limitation of this approach includes the requirement of smartphones to have Bluetooth enabled, and the need to assign a label to each beacon in order for it to be detected with My Places, which will not scale if the number of personal places is large.

Future work may consider the social aspects of personal places to automatically add beacons to My Places. Other sensing technologies, such as GPS and Wi-Fi, could be used in combination with or as an alternative to Bluetooth beacons, to increase accuracy and to detect a wider variety of places. More advanced distancing techniques will increase the opportunities to interact with a smartphone, by detecting more fine-grained areas inside a place. A rapid prototyping approach will enable developers to rapidly-prototype interactions that incorporate the context of place before committing to a more accurate positioning system.

## 4.3 Summary

This chapter presented novel rapid prototyping tools that integrate two very different approaches to detecting the smartphone user in places. Both sensors had been newly released at the time of writing, and few tools were available to integrate their sensing with the smartphone. The Microsoft Kinect depth sensor was configured to detect the smartphone user by tracking the skeleton of the user and triggering functionality when the position of the dominant hand entered a pre-recorded coordinate. Low-energy Bluetooth beacons were used to mark the coarse-grained context of a place, and functions can be triggered when the device enters a known range. Positioning experiments were performed to consider the suitability of both sensors for rapid prototyping.

The first experiment demonstrates that the Microsoft Kinect depth sensor can be used as a low-cost solution to prototype detailed interactions in a limited area of a place. A positioning experiment highlights the limitations of this approach, including the limited field-of-view, jitter in the sensor readings and occlusion. Therefore, any mobile interactions that are designed with the Kinect will require a more accurate sensing technique to be used in the wild.

The second experiment demonstrates that Low-Energy (LE) Bluetooth beacons can be used to detect coarsely defined areas, and can be used to prototype interactions that account for the context of places. An experiment was performed in a home setting to compare the characteristics of the signal received from two types of Bluetooth beacons. Though both beacons suffer from interference and low-population, the results show that the new Bluetooth LE beacons provide a more accurate approach to detecting places than traditional Bluetooth.

The rapid prototyping tools were developed to provide a sensor agnostic approach to detecting the smartphone user. Interactions that are designed with these tools are interchangeable when more accurate sensing techniques become available.

The development of rapid-prototyping tools is the first step to understanding the capabilities and limitations of a rapid-prototyping approach to position sensing (RQ-4 of Section 3.1). The following chapter will evaluate the rapid-prototyping approach with working prototypes that are developed with these tools, in an investigation of the opportunities to situate mobile interaction in physical places.

# Chapter 5

# Situating Mobile Interaction in Personal Places

The previous chapter introduced rapid-prototyping tools for sensing the smartphone user in personal places. In this chapter, potential applications of this context are explored to gain knowledge of how situating interaction might support users to manage their presence in personal places (RQ-1 of Section 3.1). First, opportunities to situate mobile interaction around artefacts of personal places are explored. Prototypes are then designed with the Microsoft Kinect to evaluate interactions with a mobile web browser around physical objects, and a digital book collection around a physical bookshelf. With Bluetooth beacons, app use is explored in personal places. Preliminary evaluations with these three prototypes are performed and the insights gained highlight the opportunities of situated interactions in personal places.

## 5.1 Finding Relationships Between Websites and Personal Places

Physical places are made up of artefacts, including gadgets, furniture and people. Some artefacts are expected to be in a place. For example, a fridge is commonly situated in a kitchen, and a desk is usually featured in an office. Many physical artefacts can be related to a digital form: maps, books and media all have a digital form. A user could interact with digital content on their smartphone by interacting around physical artefacts situated in personal places. For example, a digital photo gallery could be launched on a mobile web browser by moving towards a physical photo frame that is associated with it. As navigation between webpages typically requires typing on a small mobile keyboard [150], mobile web browsers could bridge the gap between physical artefacts and webpages to improve the user experience.

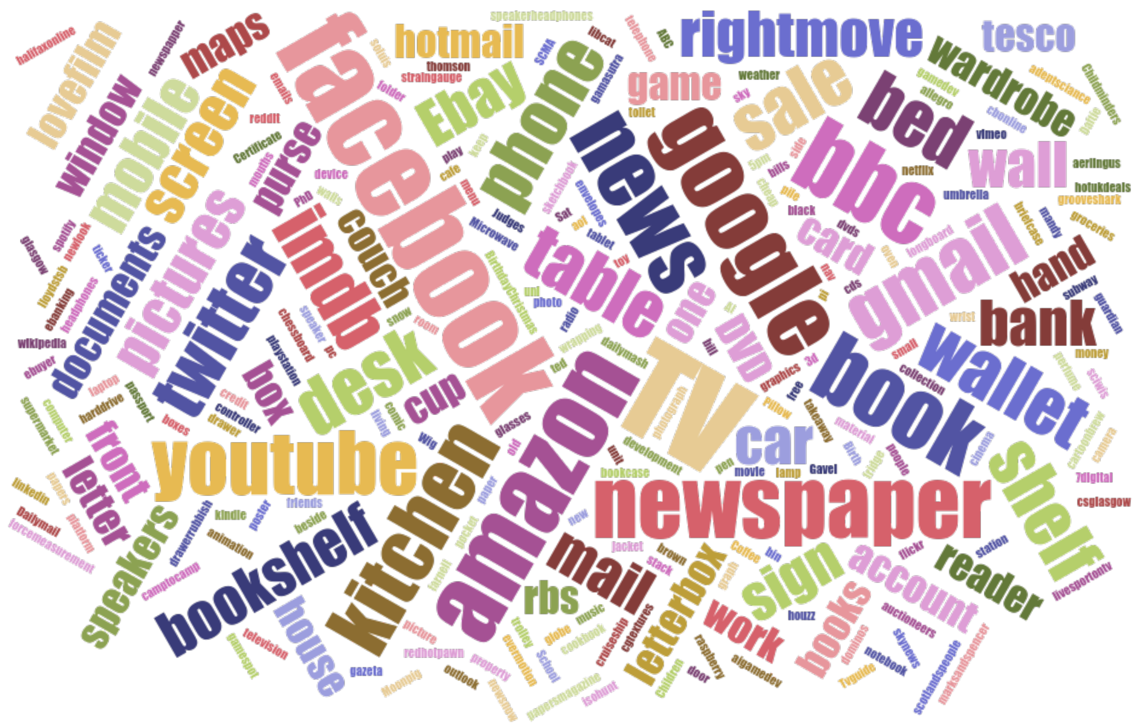



Figure 5.1: Word cloud of artefacts and websites. The top 5 are listed in Table 5.2.

In [79], modularised spatial ontologies are used to describe assisted living systems: A room is recognised as having architectural, structural, spatial and physical elements that share properties. For example, a door can be described by an architect in terms of its dimensions and where it is fixed to a wall, by a construction worker in terms the structure of how it connects rooms, a painter in terms of its colour and physical appearance or by a resident in terms of the actions they perform with it. A home automation system can be configured through these ontologies to allow a separation of concerns according to the task being automated. Blended spaces choose parts of the digital domain to blend with the physical [12].

An online questionnaire was sent to architects and computing scientists to gain inspiration for the ways that webpages could be organised around artefacts in the physical environment. This questionnaire is available as Appendix A. Architecture is the art and science of designing buildings, and an architect's profession is to design spaces to be occupied and used by humans. A computing scientist has an understanding of the capability of software. The associations suggested by each field is used to group artefacts and webpages, and consider how relationships might be formed between the two.

### 5.1.1 Questionnaire Design

The questionnaire asked users to think of 10 websites that are visited frequently and to consider an artefact for each that would best represent it in the physical environment. The

| Count | Website |
|---|---|
| 11 | Facebook |
| 7 | Amazon |
| 5 | YouTube |
| 5 | Gmail |
| 5 | BBC News |

(a) Websites.

| Count | Artefact |
|---|---|
| 12 | TV |
| 6 | Newspaper |
| 6 | Book |
| 5 | Kitchen |
| 4 | Shelf |

(b) Artefacts.

Figure 5.2: Two tables show a list of the top 5 (a) websites and (b) artefacts that were suggested in response to the questionnaire. These responses are visualised in Figure 5.1, and clustered in Table 5.3.

question used in the questionnaire was as follows: 'Write the name of a website that you visit often and an artefact that you would represent this with in your physical environment'.

The questionnaire was emailed to 60 people in total, 30 architects and 30 computing scientists. 15 anonymous submissions were received with a response rate of 25%. 8 respondents were from an architectural profession and 6 were from a computing science background. Respondents were aged between 20 - 60, and 6 were female. All persons owned a smartphone, and 60% browsed the internet on their mobile device daily. In total, 138 website and artefact pairs were gathered.

### 5.1.2 Results

A variety of websites and artefact pairs were suggested, as illustrated in the word cloud[1] in Figure 5.1, and the table in Figure 5.2. Artefacts were manually clustered into 4 categories, and websites into 9 categories, in Table 5.3 (a) and (b). Websites were grouped according to the services that they provide. For example, Amazon and Ebay are both shopping services, and YouTube and Netflix are video services. Artefacts were grouped according to their size and mobility: An envelope and a cup are **Objects** that can be moved between places. **Structures** are usually found in a single place. For example, a letterbox in a hallway, and a fridge in the kitchen. **Places** were also identified, such as a car, house or a school. Body parts and people were classified as **People**.

Table 5.3 (c) shows the top pairs of web service and artefacts in these categories. The most common example was associating a shopping service to an object, such as Amazon to a book. Video services were frequently related to structures, such as YouTube with a television.

Though the questionnaire did not ask for any reasoning, some responses included the reasons behind the relationship between an artefact and a website. One respondent acknowledged

[1] `http://www.jasondavies.com/wordcloud/`

| Service | No. | Examples |
|---|---|---|
| Shopping | 31 | *Amazon, Ebay, M&S* |
| Search | 22 | *Google, IMDB, Weather* |
| Reader | 19 | *News, Reddit, Gamespot* |
| Social | 18 | *Twitter, Facebook, LinkedIn* |
| Video | 14 | *YouTube, Netflix, Vimeo* |
| Email | 13 | *Gmail, Hotmail, Yahoo!* |
| Work | 8 | *Company website* |
| Banking | 8 | *RBS, Halifax* |
| Music | 3 | *7digital, Spotify* |

(a) Web Services.

| Artefact | No. | Examples |
|---|---|---|
| Object | 80 | Envelope, Book, Cup |
| Structure | 48 | Desk, Wardrobe, TV, Wall |
| People | 6 | Hand, Mouth, People |
| Place | 4 | School, Car, Kitchen |

(b) Artefacts.

| Service | Artefact | No. | Example | |
|---|---|---|---|---|
| Shopping | Object | 18 | *Amazon* | Book |
| Reader | Object | 13 | *Reddit* | Newspaper |
| Search | Object | 13 | *Google* | Cup |
| Video | Structure | 11 | *Netflix* | Television |
| Shopping | Structure | 10 | *M&S* | Wardrobe |
| Social | Object | 10 | *Facebook* | Photos |
| Search | Structure | 9 | *IMDB* | Poster on wall |
| Email | Object | 8 | *Gmail* | Letter |
| Banking | Object | 7 | *RBS* | Bank card |
| Reader | Structure | 5 | *BBC News* | Window |

(c) Top 10 Service - Artefact pairs.

Figure 5.3: Services and Artefacts.

their frequent use of Facebook, and chose to relate this social networking app to their “hand (since it is always around, and I seem to always check Facebook)”. Google, the search engine, was related to a “cup on table - filling up with items. or drawer/rubbish bin representing searching”, and BBC News was related to a “window [to the outside world]”. Location-dependent options were considered, for example relating Netflix to “whatever screen is on front of me - be it my television or laptop screen (depending on where I am)”. Temporal relationships were also specified, for example New Look, a shopping service, to “My new jacket”. Some responses indicated structures in certain places, for example relating Cartoon Brew, a video service, to the “shelf where I keep my animation DVDs” and GameDev, a work resource, to a “game development bookshelf”.

### 5.1.3 Discussion

The variety of website and artefacts specified in the questionnaire demonstrates that mobile users can be creative when relating to the physical environment. By grouping these artefacts and websites into categories, common themes could be identified. Different sensors will be

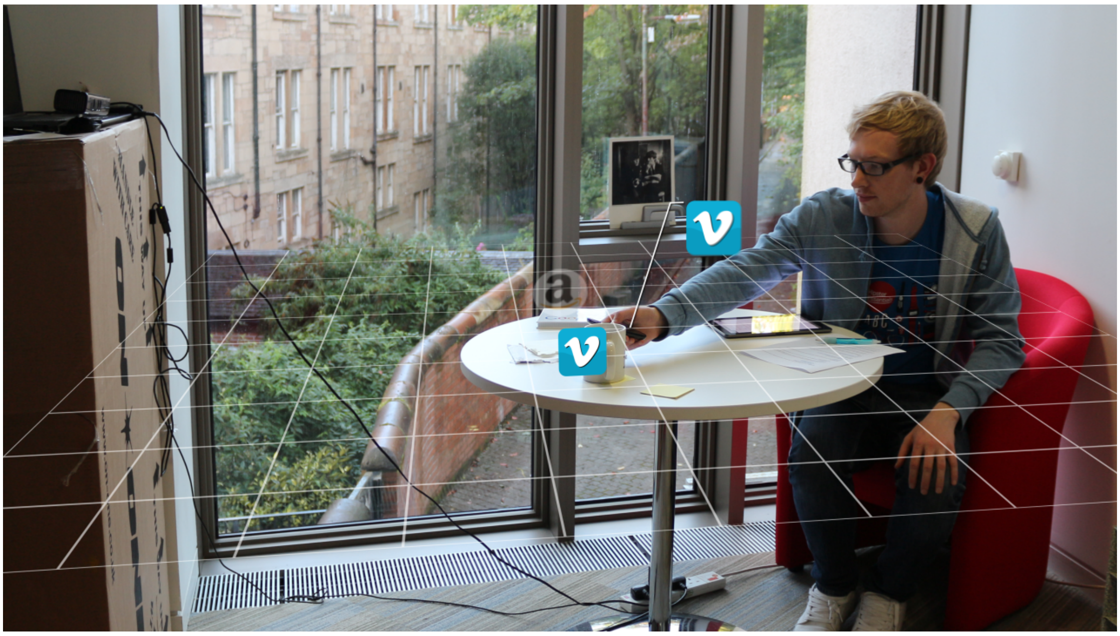


Figure 5.4: Assigning a webpage to the coffee cup by holding a smartphone next to it.

required to detect each artefact group. The short range of an NFC tag might be appropriate to keep track of the small, well-defined area of an object. Bluetooth might be appropriate for marking a place with less well-defined boundaries. Of the artefact groups that were identified, objects were most frequently associated with websites. This questionnaire allowed us to gain insight and grounding from smartphone users about how web services relate to the physical environment.

## 5.2 Bookmarking Websites with Physical Objects

Smartphone users can access a wealth of information at their fingertips through mobile web browsers. However, web browsing suffers due to the small mobile keyboard and the need for users to recall the Uniform Resource Locator (URL) [150]. It is important to explore new input techniques for accessing information on a mobile device to improve the user experience of mobile web browsing. Tagging objects in the physical environment with websites can be considered to link to a mobile web browser. An Amazon Kindle could relate to the online book repository, and a physical newspaper could relate to its online counterpart. Bridging the gap between the digital and the physical can create new opportunities to interact with a mobile device. However, is not clear how smartphone users will perform interactions situated in the physical environment.

Augmented reality applications rely heavily on the visual representation of information in a digital overlay [118]. In contrast, imaginary controls can be positioned in physical space, and properties of the physical environment can support the visio-spatial memory to interact with a mobile device. An imaginary interface [63] demonstrates that the visio-spatial memory can replace visual feedback for a non-visual mobile application, by situating the interface around the non-dominant hand. Spatial information has been explored in mobile applications by Strachan *et al.* in [142] as bearing-based interaction, that can negotiate the selection of targets at a distance: Users can probe and query information in a geographic space, and information can be accessed by skilful pointing of the device towards the expected location. However, location-based positioning is restricted to scenarios of outdoor locations. Instead, information can be organised around physical artefacts in indoor places. The navigation of menu shortcuts in egocentric space is demonstrated with Virtual Shelves [93]: the smartphone can be pointed towards sectioned areas around the user to explore the menu. Users can associate application shortcuts with each section. This technique could be used in exocentric space: the Microsoft Kinect can be used as an external positioning system, and the field of view can be sectioned into slots.

A prototype is designed that enables objects in the physical environment to be augmented with digital information, and spatial interaction is used to explore content. Using the Kinect and Mobile Interaction Tool from Section 4.1, information is be placed and accessed by moving the mobile device towards objects in each slot. In the evaluation below, two scenarios are used where spatial web browsing could fit in to an office routine. The spatial web browser integrates with the Kinect to enable objects to be tagged with websites. A discussion is provided on the way that participants approached the objects they interacted with using the smartphone, and the relationships that participants formed between websites and objects.

### 5.2.1 Prototype Design: Mobile Web Browser

The mobile web browser extends the Android WebView with position sensing of a Microsoft Kinect motion sensor, by communicating with the Mobile and Kinect Interaction Manager tool from Section 4.1, as displayed in Figure 5.5. This tool is based on the Virtual Sensor approach demonstrated in [112]. The Kinect tracks the dominant hand of a person in its field of view, and this can be used to simulate the tracking of a smartphone that is held. To track the position of the hand, the Kinect software required the Psi calibration pose to be performed, a stance where the arms are held at a 90° towards the Kinect.

The web browsing experience is augmented by automatically loading pages when the device is moved into a position that has been tagged with a webpage. A SlidingDrawer was used to present the spatial bookmarks in the web browser, which could be opened and closed by dragging up or down from the arrow icons displayed in Figure 5.6.

To avoid unintentionally loading a bookmark, the web browser only updates the webpage if

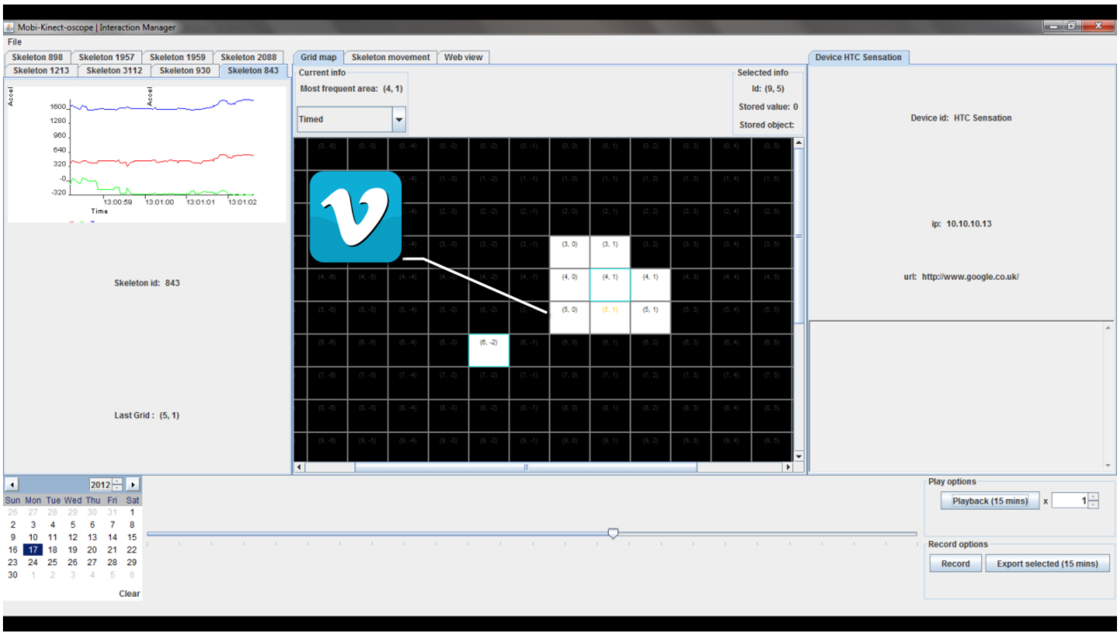


Figure 5.5: Interactions with the mobile web browser can be recorded in the Mobile and Kinect Interaction Manager. Web pages can be stored as a hotspot defined as an (x, z)-position, which triggers a command to open the web page on the smartphone and in the tabbed pane of the evaluation tool.

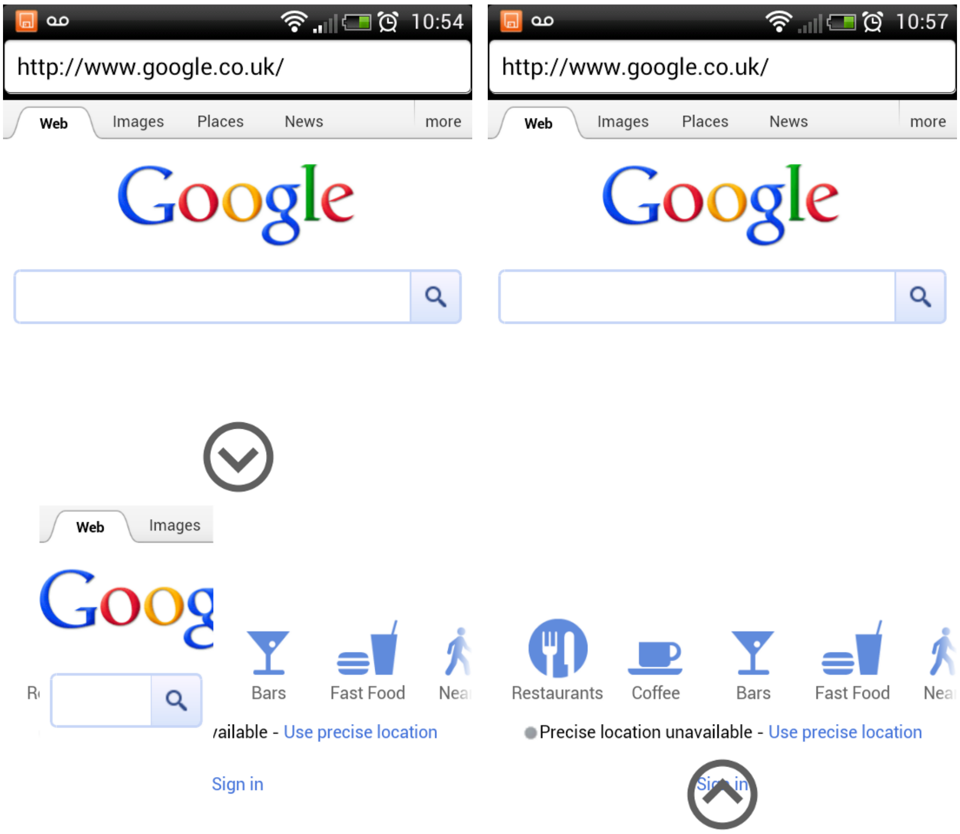
the drawer is open. If the drawer is closed when the smartphone is positioned near a tag, the bookmark in the drawer updates, such that it can be clicked when the drawer is next opened.

To assign a webpage to an artefact at a position, the sliding drawer is opened, and the screen is long-pressed. This sends a signal to the Mobile and Kinect Interaction Manager, and replaces any previously stored URL. When the hand position enters the proximity of this position again, the URL is sent to the device. The Mobile and Kinect Interaction Manager also loads the webpage, to allow monitoring of what is displayed on the private mobile display during the evaluation.

The Kinect was positioned 1 metre from a desk, and its sensing space was split into a 11 x 9 x 2 grid with a divider at 153cm(h), allowing webpages to be stored in 40cm(w) x 35cm(d) areas around the desk. The thresholds for these dimensions were found experimentally, and required a balance between the sensitivity and resolution of the space. This grid is visualised in the Mobile and Kinect Interaction Manager screenshot in Figure 5.5.

## 5.2.2 Evaluation: Object Tagging with a Mobile Web Browser

An evaluation was performed with the spatial web browser, that asked participants to take on the role of someone who frequently visits websites on a smartphone while sitting at a desk.




(a) Open. (b) Closed.

Figure 5.6: The spatial web browser prototype. A drawer is overlaid on top of the webpage, which can be opened (Left) or closed (Right) by clicking on the arrow icon. Screenshots of bookmarked webpages are displayed in the drawer, and update automatically as the smartphone moves around tagged physical objects.

### Research Method

The lab environment was set up as an office. Participants sat at a desk, and objects placed around the desk were in view of the Kinect. On the desk, there were sticky post-it notes, a mug, a pile of envelopes, a notepad, a Samsung Galaxy tablet and a pen. At the window to the right of the desk was a picture. To the left of the desk was an organiser containing CDs, and a stack of books sat on top. The set up of the evaluation, displayed in Figure 5.7, remained consistent between participants.

Two scenarios were presented, that call for a website to be associated with an object on the desk. The order of the scenarios alternated between participants to avoid a bias in the selection of objects. The scenarios were as follows:

Figure 5.7: Experimental set up.

1. On a coffee break, John likes to check if there are any new Romance novels that he should buy from Amazon.
   The URL to this page is: *www.amazon.co.uk/Romance-Books/b?node=88*

2. Marie likes to start the day by watching a new video on the Vimeo robot channel.
   The URL to this page is: *www.vimeo.com/channels/robots*

Before the main evaluation task, participants were given a brief introduction to associating and retrieving URLs using the spatial browser by performing a short training task. Participants were not given instruction on how to hold the smartphone to perform an association between the webpage and an object.

### Task

The evaluation task was to assign the URL in the scenario to an object on the desk using the spatial web browser. After assigning both URLs, participants were asked to fill out a questionnaire about the objects that they chose. This questionnaire is available as Appendix B. Requesting participants to complete the questionnaire between the association and retrieval tasks ensured a short break between these stages, which is more usual of normal browsing behaviour. Finally, participants were asked to retrieve each URL that they bookmarked by

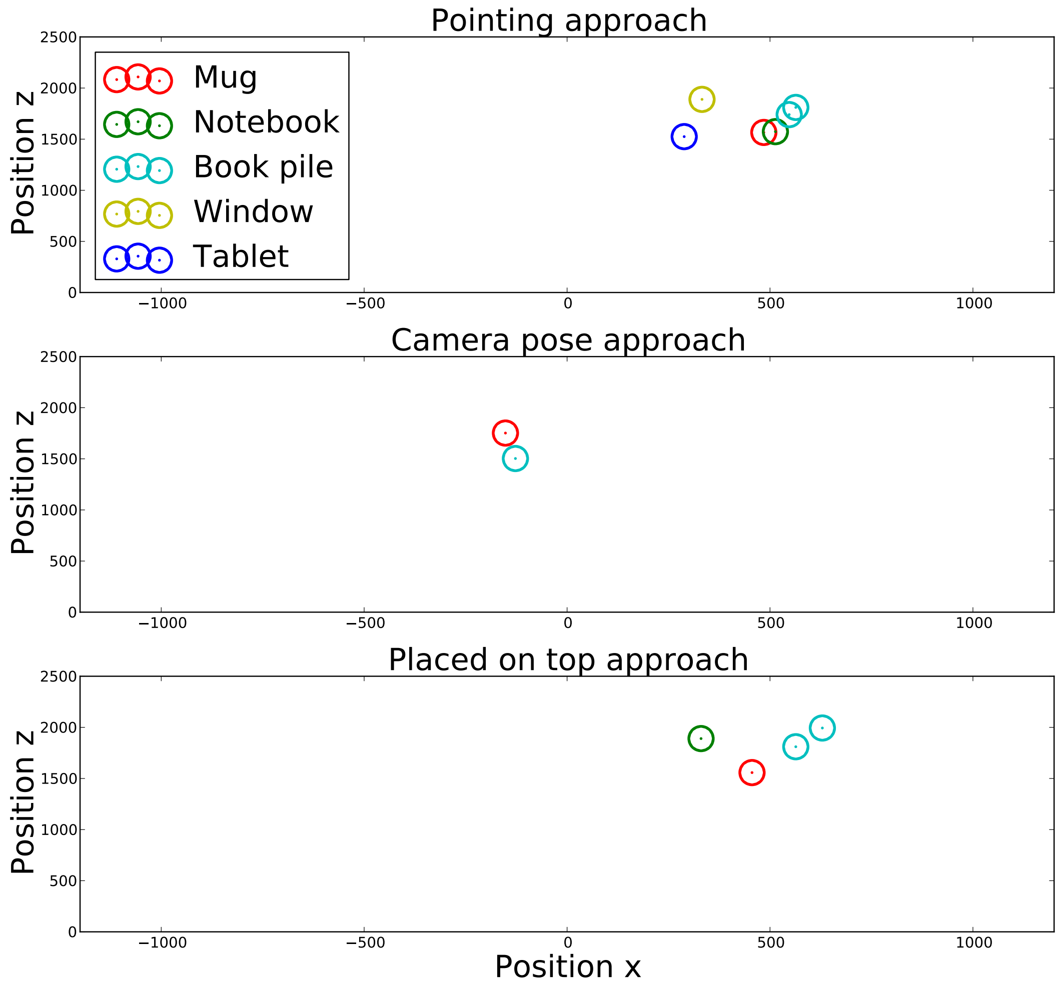



Figure 5.8: 12 positions associated by 6 participants for Scenarios 1 and 2. Three approaches to tagging objects were identified.

recalling the object for that URL, and positioning the smartphone in its position. A final questionnaire asked about the experience of retrieving the URL.

### Participants

7 participants, who were recruited as students from the University of Glasgow, took part in the evaluation. Participants were all male, from a technical background, and aged between 20 - 50. Participants stated that they were right handed, and chose to operate the mobile device with their right hand.

### 5.2.3 Results

During the evaluation, it was not possible to record the positions of one participant, due to difficulty with the Kinect tracking. Figure 5.8 displays the positions of webpages selected by the remaining 6 participants.

#### Gestures

Participants were observed to approach the objects with the smartphone in different ways. Four participants held the smartphone with one hand and long-pressed with their thumb, such that both their hand and the device were pointing at the object. Two participants were observed to place the smartphone on top of the object, long-press with their index finger, and move their hand away before picking up the device again. One participant pointed the back-facing camera of the device in the direction of the object, and long-pressed as through they were taking a camera photo. All participants repeated the gesture that they performed to retrieve the website from the object, and the system responded appropriately.

#### Relationship Pairs

Participants reported different reasons for choosing objects, including: the context of the scenario (e.g. coffee break), the service provided by the website (e.g. books), and the subject of the website (e.g. romance). Four participants assigned the Romance novels URL to the pile of books, and two participants assigned this URL with the coffee cup. P1 “placed the website on the coffee mug because it was being retrieved on [a] coffee break.”, whereas P4 states their reason to be “since [the] url was about books”. Two participants assigned the Robot video URL to the coffee cup. P2 explained his choice to be related to his personal morning routine, as his “first morning task is [to drink] coffee [and] watching the video [is what he would do] at the same time”. P3 chose to assign the website to the power socket on the wall because “robots are electrical” and the “book pile was next to the power plug”. P1 “placed the website on the tablet screen since it was for a video”.

Some participants were dissatisfied with the selection of objects available, and so they chose an object that they thought was close or distinctive. The window was chosen since “it was near to my right hand” [P4], the book stack as it was distinct enough to recall with “short term memory” [P5] and P7 “chose the notepad as it was an obvious object that was close to hand”. Participants wanted more objects to choose from, for example P5 “wanted something with romance to relate romance novels to”. P1 said that they would “remember where I had put sites better if I was doing it with my own stuff”.

### Limitations

P8 was unsure about the use of movement for interacting with their device, and was "not sure I'd ever use this in practice. Basically I'm too lazy to use this over a more traditional bookmark list.". Participants were also concerned about how the spatial interface would scale. P6 was "not sure if they would be the same if number of artefacts was higher e.g. 10." [P6]. Additionally, P3 "would remember pages I use often that are part of my routine (every morning, every weekend). For more rare actions, buy train ticket for example, I would have trouble to remember". Similarly, P5 shares, "This novel interaction method is very natural and easy to learn. I believe I could benefit from it in real-life scenarios. However, I am not confident enough about memorising too many artefacts and the associated bookmarks". This prototype would need to be tested in the wild to better understand the limits of a spatial web browser.

## 5.2.4 Discussion

A sensor-agnostic approach was presented to designing interactions with a smartphone around objects in the environment. By prototyping the spatial interaction with the Kinect, it was possible to quickly test the prototype and explore the interaction ideas for tagging objects with websites. The results demonstrate different ways that smartphone users can associate webpages to objects with a spatial web browser. Participants were able to relate webpages to objects, and recall these associations to navigate the mobile web browser.

By using the Microsoft Kinect to simulate positioning, different ways of approaching an object with a mobile device were observed. It was not possible to record and compare the positions selected by each participant. Despite the different approaches to tagging objects, participants were able to rediscover information that they placed in the spatial environment by using consistent gestures to place and retrieve webpages. Therefore, it was sufficient to track the dominant hand of the participant in order to relate this position to the movement of the mobile device.

Several limitations were reported that were experienced when using the Kinect. This study was limited to fixed positions that were in clear view of the Kinect. Participants were partly occluded behind a desk, and this caused the Kinect tracking to be unreliable. The precision of the positioning was limited due to jitter in the Kinect tracking. It was also not possible to track multiple users without disambiguation between Kinect skeletons and the mobile device.

In this study, participants only located URLs positioned by themselves, but not those bookmarked by other people. Future work should consider social issues when multiple users sharing objects, and the opportunities to explore the public and private nature of physical

Figure 5.9: Standing near a physical bookshelf to browse digital books on a smartphone.

objects. A spatial interface could invite others to browse webpages associated with objects on display. Alternatively, webpages may be hidden, such that its position is known only in the mind of the user. Though this study is limited to a small number of participants and scenarios, it demonstrates that participants relate to the physical environment in different ways. Personalisation will be important in the design of interactions situated around physical objects.

## 5.3 Putting Books Back on the Shelf

E-book readers (*e-readers*), such as Amazon's Kindle, are a popular way to consume hundreds of books without the clutter of their physical form. Although the sale of e-books has overtaken physical books on Amazon,[2] digital books are often used interchangeably with physical ones [130]. One reason why e-readers have not fully replaced paper books is that they constrain interactions to its private display. Therefore, they lack the rich user experience that is expected from physical books. Apart from unique haptic experiences, such as annotating a page or marking it with a dog ear, physical books afford the ability to be organised in the context where they are used. For example, one might keep recipe books on a different

[2]Amazon Press Release, August 2012:
`http://phx.corporate-ir.net/phoenix.zhtml?c=251199&p=irol-newsArticle&ID=1722449`

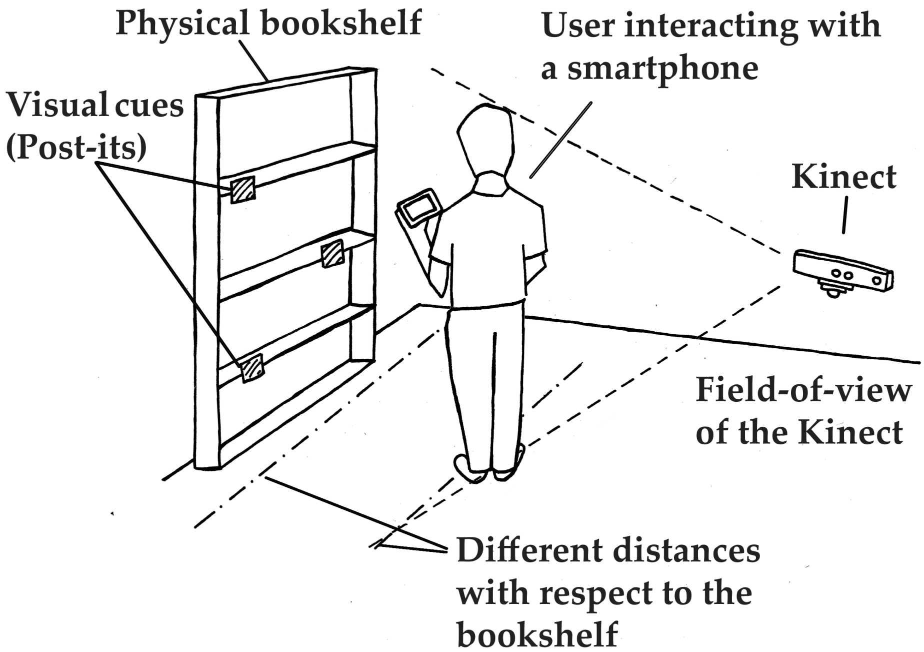



Figure 5.10: A user stands at a section of the bookshelf and views a collection of e-books on his smartphone.

shelf or in a different room than science fiction novels. Familiarity of the physical layout can be used to find where each collection is. This has the additional effect of making interactions with books visible to others.

In 1995, Kirsh considered how the spatial arrangement of artefacts in a physical environment can simplify choice and perception [87]. Similar to his approach, the cognitive load of mobile applications could be reduced by utilising relationships to spatial layouts. This work is also inspired by 'situated information spaces' [54], a term that describes the principle of anchoring digital information to physical objects so that it may be browsed and manipulated in the context in which it originated. This is a converse approach to the 'Internet-of-Things' [89], which is concerned with indexing physical objects in the digital world. The 'Bohemian Bookshelf' [149] considered the visual presentation of digital book collections as an information visualisation. The 'Digital Bookshelf' [4] demonstrated a way to present e-books using a projector in combination with the Microsoft Kinect, a depth sensor that provides natural interaction and has been recognised as a platform for prototyping interactions in combination with smartphones [112]. The distance of a user to a public display has been explored in combination with physical artefacts as a spatial interaction technique to automatically adjust information displayed on the visual interface [7]. In comparison to proxemic interaction, the visual properties of the sensing space can be utilised as a natural way of interacting with information on a mobile device.

As a step towards improving the user experience of e-readers, digital books can be integrated

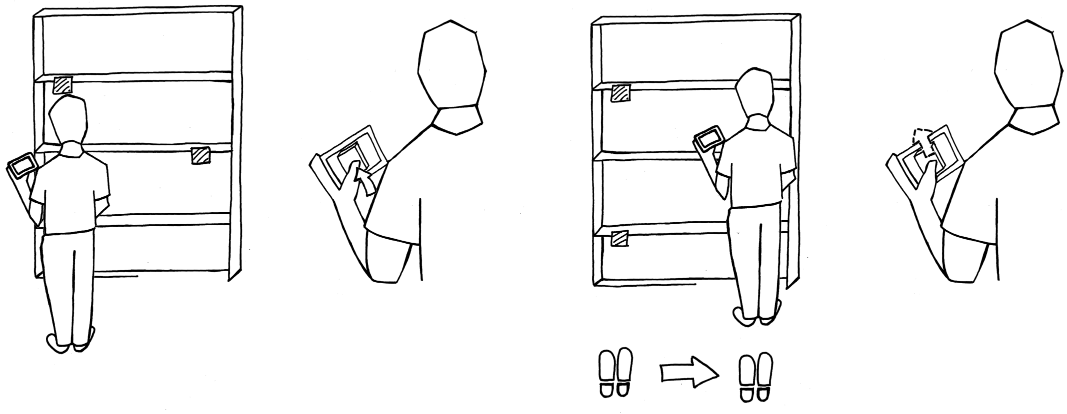


Figure 5.11: 'Drag-and-drop' metaphor (left to right): A user stands at a section of the bookshelf, *grasps* an e-book, *drags* it to a different section and *drops* it.

into the structures that exist in the physical environment. A prototype is designed that uses the Kinect to detect movement between sections of a physical bookshelf, and a mobile app is developed that allows digital books to be explored by moving between each section, illustrated by Figure 5.10. Though this prototype situates interactions around a single bookshelf, one could imagine spreading digital book collections in multiple rooms.

### 5.3.1 Prototype Design: Digital Bookshelf

The prototype consists of three parts: 1) a physical bookshelf that has been labelled with handwritten notes that act as visual cues for the digital book collections, 2) an Android application that displays the digital books, and 3) a positioning application that uses the Kinect placed 1 meter behind the user to detect when their center of gravity enters a section, and communicates this to the mobile device via a Wi-Fi network.

While there are several approaches to detect the position of a user in relation to a physical artefact, the Kinect [4, 112] was favoured over tag-based [69] or capacitive sensing [162] approaches. This provided freedom to anchor book collections anywhere on the bookshelf without limiting the design to the number and placement of tags or to the size of the capacitive surface.

The mobile app was built on top of an existing prototype developed by a masters student, which displays 3D models of physical books in a scrollable, horizontal list. A screenshot of this app is shown in Figure 5.9. Compared to displaying book titles in a 2D grid or list, the app simulates 3D books on a physical bookshelf, which can be looked at closer by tapping on the spine of the 3D model. This app was chosen to explore interactions with digital books that are interweaved with physical books on a bookshelf.

Interactions were designed to browse between collections of e-books (*filter*), gain an overview of all sections (*take a step back*) and organise e-books into categories (*drag-and-drop*). The 'filtering' metaphor extends the physical action of standing in front of a section of a bookshelf to zoom in on a particular category, by filtering the mobile app for books in this collection. 'Taking a step back' and increasing the distance to the bookshelf displays all collections and provides an overview of what is there. The 'drag-and-drop' metaphor allows digital books to be moved between collections by dwelling a finger on the e-book (to 'grasp' it), walking to a new area of the bookshelf ('drag') and then letting go ('drop'). This is illustrated in Figure 5.11. Tactile feedback is provided on the mobile phone to indicate when the user filtered a collection, when a book was selected and when a book was successfully placed in a collection.

### 5.3.2 Evaluation: Browsing Digital Books with Movement

An evaluation was performed to gain feedback on the design of the digital bookshelf prototype.

#### Participants

9 participants, aged between 21 and 30 and with a technical background, were invited to find and categorise computing science books in an office environment.

#### Task

The search task required participants to select a sequence of e-books. The name of the book and its category was displayed on the screen, and the participant used the appropriate menu to find book. Once the correct book was selected, the participant was notified and the next book to find was displayed.

The categorisation task required participants to move e-books from one collection to another. Similar to the search task, the name of the book and its category was displayed on the screen, along with the category that it should be moved to, and the participant used the appropriate menu to find book. Once the correct book was categorised, the participant was notified and the next book to find was displayed.

#### Procedure

Each task was performed 10 times using the 'filter' and 'drag-and-drop' situated interactions, and again using a menu interface on the device alone. After the tasks, a discussion was led to gain qualitative feedback about the prototype design.

### 5.3.3 Results

In general, all participants were able to relate the sections of the physical bookshelf to the mobile interface and use these to find and organise digital books on the smartphone. P2 considered the situated interface to be useful for accessing collections and imagined that he could “go to it and say, there are all my work books, and then bang they’d all be there, and then I’d just be able to extract them like I normally would”.

The handwritten notes were considered to be part of the interface: P4 “always looked at the post-it [...] even if [the category] was displayed on the device”. This suggests that the artefacts should be clearly marked and discoverable. This also highlights the importance of keeping the position of artefacts consistent with the digital information associated with them. This would be particularly important if the artefacts could be moved. For example, if bookends were used to mark each section, then the position of the bookends would move depending on the number of physical books present on the shelf. In this case, their position might be better tracked using tags instead of the Kinect.

Participants were interested in the potential to incorporate their digital collections into their interior design, and “not wanting to have a whole bookshelf taking up things, and knowing that this bit [...] is where I keep all my books and this is where I keep all my work books” [P2]. An example of using wallpaper to design a situated bookshelf is shown in Figure 5.12.

Participants also expressed the desire to personalise the artefacts. P4 stated that “If I knew that I kept my pictures in a corner then that would be enough for me to go over and see them on my phone”. Personalisation might explicitly discourage discoverability, by requiring external viewers to know the physical layout before they can make inferences about how a person is interacting with their mobile device. Additionally, one could be aware of another person browsing ones personal collections if they were situated in a personal place. This could be valuable as a tool that prompts appropriate social behaviour.

In the experimental setting, movement was perceived to be stimulating, and more natural to control the book collections than navigating a menu interface. P3 found that “even though it is more physical it is less boring”. However, participants could imagine times when they would and would not want to move around to interact with their device. P1 considered, “If I was sitting down [...] then I would just use the menus [but] if I couldn’t find it then I would use [the situated interface]”. Therefore, situated interactions should be considered as supplementary to the menu interface, and provide value in certain circumstances.

Figure 5.12: Interior design can be considered to organise digital book collections on a smartphone.

### 5.3.4 Discussion

Situated interactions with a smartphone around a physical bookshelf were investigated as a way to improve the user experience of finding and organising digital books. The digital bookshelf prototype uses the Kinect depth sensor to detect the position of a user in relation to sections of a physical bookshelf. A mobile application was developed that allows the user to browse and organise digital books by moving between physical sections. Initial observations were presented from a user study that evaluated the search and categorisation tasks with this prototype. The findings motivate reasons to explore digital books in a physical environment, and indicate issues to consider when designing situated interactions with book readers.

This investigation is a first step in exploring how situated interactions with digital book collections could enrich the use of e-readers. The next step is to explore the impact of this design on the social aspects of situated interactions and the extent to which structures in the physical environment can be used to represent elements of the mobile interface.

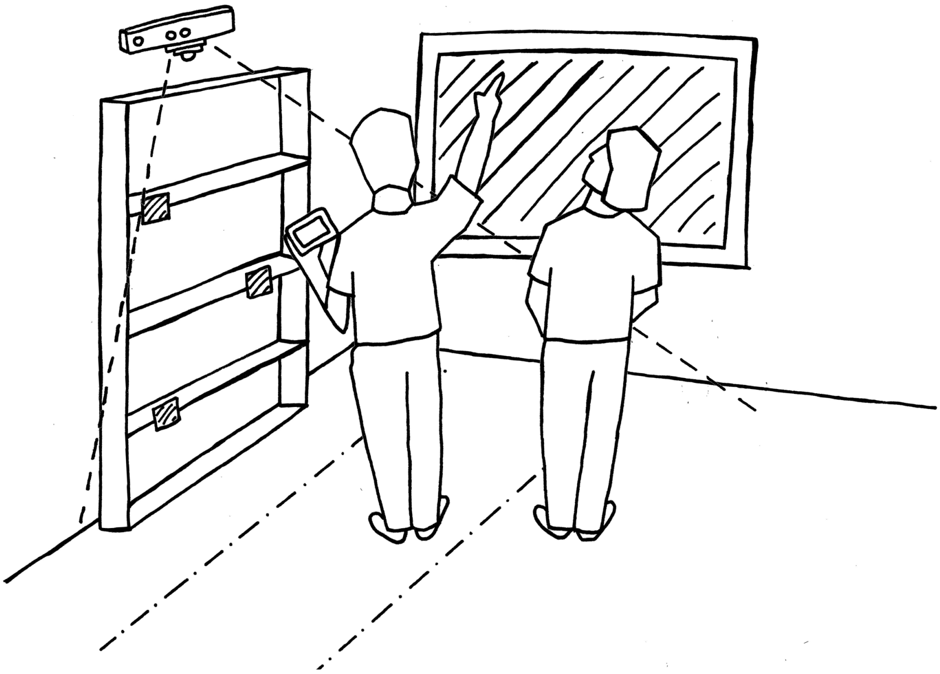


Figure 5.13: Two users stand at a section of a room and interact with a smartphone.

## 5.4 App Tracking in Places

Tracking app use in relation to contextual information can be valuable to gain insights, such as where apps are used in the wild [16]. App tracking can also be useful for building applications on top of usage data, including context-aware recommendation systems [1, 136, 168, 37]. However, it can be difficult to track mobile information needs in personal places. So far, studies that collect information needs in personal places have been limited to diary studies [28, 30, 38, 141, 77, 153] or coarse-grained geographic locations [85, 153].

A study of application usage on Android mobile phones was presented in [16], and a dataset was gathered from users of an Appazaar context-aware recommender application. In this study, app usage was analysed in terms of chains of application usage and also in terms of context, namely time and geographic location, which helped them improve the prediction accuracy of mobile application recommendations. Similarly, Applause [37] is a context-aware recommendation system that tracks the location of app installs and recommends apps for other users to install in a given location. A predictive approach to designing an adaptive homescreen was taken by [138]. However, as the predictive algorithm learned from multiple contextual sources, participants were unable to follow the updates the dynamic homescreen and found this to be frustrating.

A rapid-prototyping approach to detecting personal places with Bluetooth beacons is taken to enable an app tracker to be developed that automatically records app use in personal places. Quantitative data is collected to gain insights into app use in more fine-grained places, and an application is built on top, that uses a record of app use in places to adapt a menu of

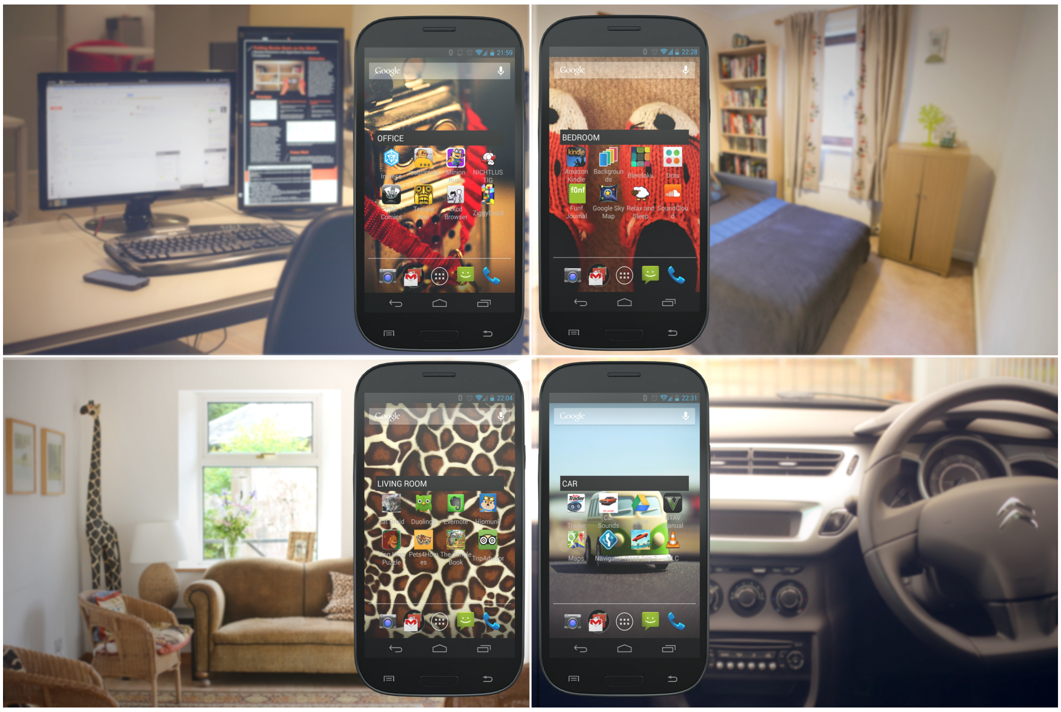



Figure 5.14: Apps can be related to the places where they are launched on a smartphone.

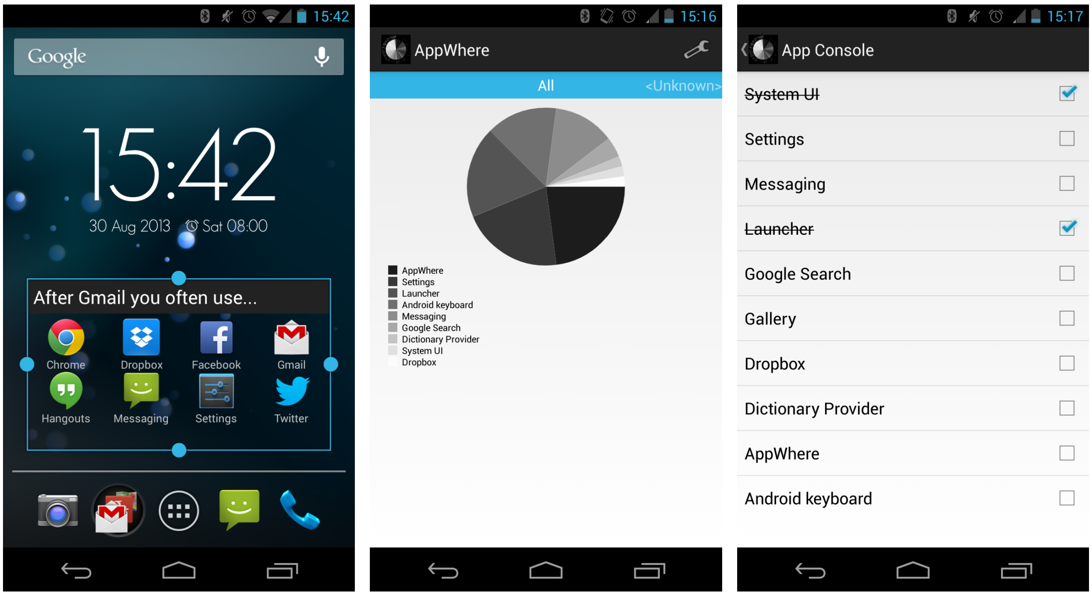
app shortcuts to the places where apps are used most. Appwhere, an adaptive homescreen menu, is designed to adapt to the context of personal places. Simple models of app use in places are explored to make updates to the adaptive menu easy to follow: the most frequently used apps, most recently used apps, and sequentially used apps (i.e. most frequently used next). Statistics that summarise app use in places are presented in the Appwhere prototype to provide awareness of app use. An interactive visualisation is also designed for researchers to gain insights into app use data in the context of personal places. Traditional Bluetooth beacons are used to demonstrate the design of a place-aware app tracker, and insights gained from a user study with 6 participants over 4 weeks are presented.

### 5.4.1 Prototype Design: Appwhere 1.0

The Appwhere prototype consists of multiple parts: place detection, app tracking, adaptive homescreen menu, app use statistics and an interactive visualisation.

#### Place Detection

My Places integrates with Appwhere to provide place detection, as described in Section 4.2. As this study was conducted prior to LE beacons, the approach is explored with the traditional Bluetooth beacons displayed in Figure 5.16. To conserve battery power, the My




(a) Adaptive homescreen widget. (b) App use statistics in places. (c) Exclude apps

Figure 5.15: Appwhere consists of an adaptive widget in the lower section of a home screen (a), statistics about app usage in places (b) and a menu to exclude apps from the app tracker.

Places tool was configured to perform Bluetooth discoveries with an interval of 20s. The place estimate is considered to be the beacon with the lowest $|RSSI|$ less than $|-70|$. As a Bluetooth discovery could take as long as 12s, an early place update is sent if a beacon is discovered with an $|RSSI|$ less than $|-55|$. These thresholds were found experimentally in Section 4.2, and the system was accurate enough for rapid prototyping. When My Places discovered a new place, a notification was presented in the notification bar to provide an opportunity to validate the place detection. If Appwhere does not receive a place estimate for longer than 5 minutes, the current place estimate is considered to be stale and the device to be in an 'Unknown' place. In addition to the app launch history, Appwhere recorded data for evaluation purposes, including whether an app was launched from the adaptive widget, and a trace of the orientation and accelerometer sensors, and the Wi-Fi and Bluetooth scans.

### App Tracking

The app tracker scans for changes to the foreground app every 1 second, and stores the name and timestamp of the foreground app in a database on the mobile device. There are two main issues with periodically scanning for apps in use: scanning this list will consume power, and so it is desirable to scan as few times as necessary. However, if scans are not frequent enough, then an app could be opened and closed before it is possible to detect it. Therefore, the minimum time for an app to be opened should be considered for when it counts as an app launch, or when the app launch was accidental. In the adaptive widget, accidental app launches will not be useful when ranking the most used apps on the homescreen. By only scanning the foreground app every 1 second, Appwhere considers apps that are used for less

Figure 5.16: SHAKE SK6, SHAKE SK7, JAKE and USB Bluetooth beacons.

than 1 second to be insignificant. Further investigation into the significance of the duration of an app launch for an adaptive homescreen may be interesting for future work.

### Adaptive Homescreen Menu

The adaptive homescreen was implemented as a widget that can be placed on the homescreen of an Android 4.0+ device, as displayed in Figure 5.15 (a). This is intended to be a useful tool for finding apps that are likely to be launched in the current place. The text at the top of the widget reflects the label of the place. When the user is detected to be in a new place, the database of app launches is queried to retrieve the ranking of apps, and the widget is updated. The widget also updates after each app is launched so that it always reflects the most used apps. Though it is possible to create predictive models of app use [138], Appwhere is explored with simple models: most frequently used, most recently used, and most frequently used next, i.e. sequentially used. The most frequently used apps can be ranked by the number of times that they have been launched in each place. Most recently used apps can be ranked by the timestamp of when they were last used in a place. The most frequently used next apps can be ranked by the number of times that they have been launched in each place after the previous app. A single ranking is chosen for the widget, and the top 6 apps are presented alphabetically.

### Statistics

Appwhere provides statistics in an app to reflect on where apps are used most. A pie chart represents the number of times each app has been launched in places, as displayed in Figure 5.15 (b). Swiping the interface left or right will present a view of app use in each place. If there was any apps that participants were uncomfortable with sharing, they could set this in the menu shown in Figure 5.15 (c) by clicking on the spanner icon. This would remove the app from both the widget and the statistics.


Figure 5.17: Interactive visualisation of app usage with traditional Bluetooth beacons. The visualisation is filtered to show the data of participants 1 - 8 in known places. The histogram indicates that participant 2 launched apps most in places, and a surge in app use occurred just before 8am.

### Interactive Visualisation

An offline interactive visualisation was created to explore and discover insights in the app use data. This is displayed in Figure 5.17. The visualisation was implemented with D3.js[3], which allows the data to be filtered in real-time. This provides an overview of the app launch behaviour, including the time of day, places, and days when app use was most frequent. App launch history could be loaded into this visualisation once it had been pre-processed.

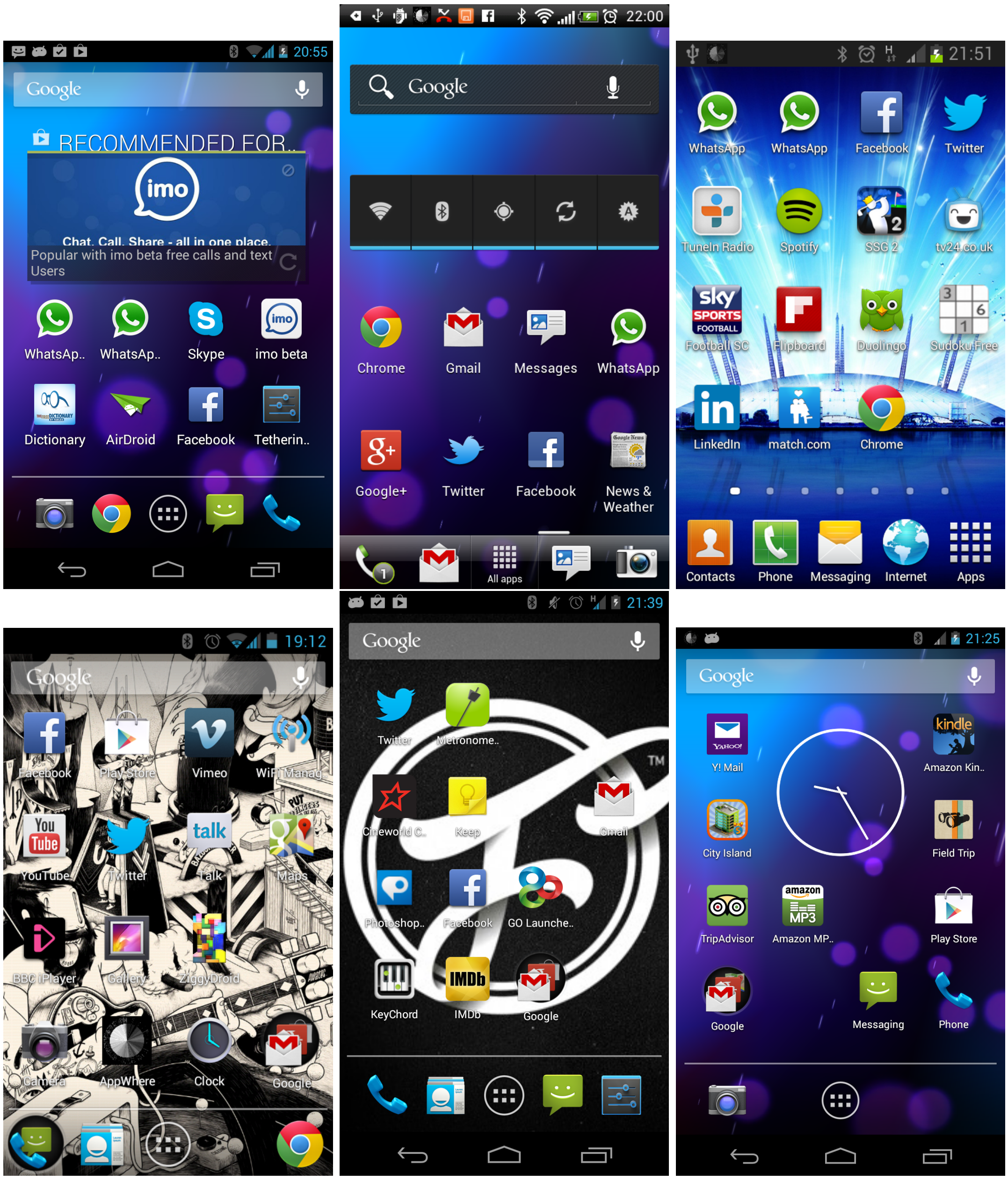



Figure 5.18: Screenshots of participant homescreens prior to installing the Appwhere widget. Many apps were placed on the homescreens, and some are duplicates.

## 5.4.2 User Study: Exploring App Tracking in Places

A user study was run to gain insight into the use of apps in places. Appwhere was installed on the Android smartphones of 6 participants, and their launch history was recorded over a period of 4 weeks.

[3] http://d3js.org/

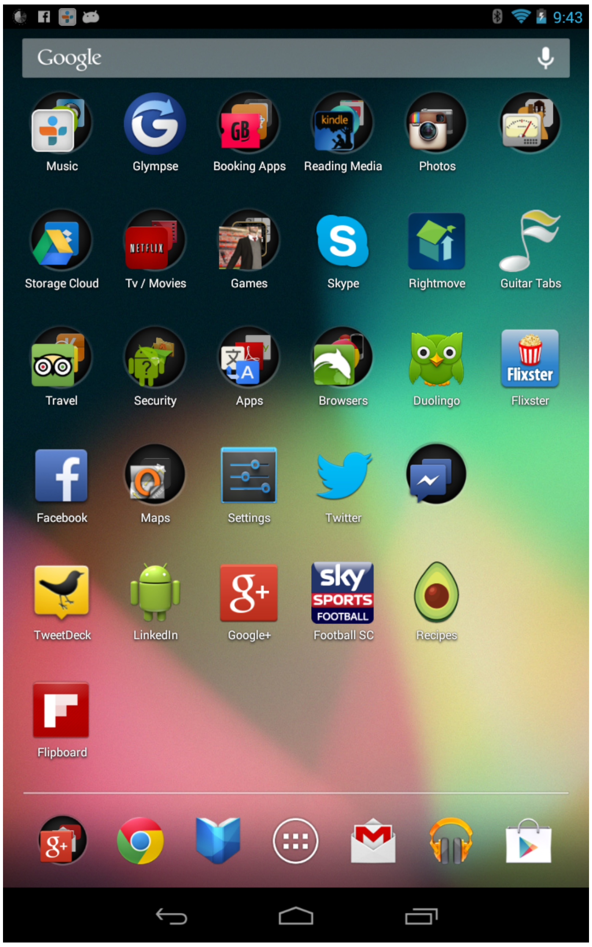



Figure 5.19: A tablet has more space for app shortcuts.

## Participants

Participants were asked to fill in a preliminary questionnaire to understand any potential source of bias, available as Appendix C: Participants were aged 27 - 61, and two were female. Two participants were from an education background, two were design professionals, and two were technical engineers. All participants had experience with an Android smartphone for at least 6 months, and owned different smartphone models with the Android 4.0+ operative system: two Nexus 4 smartphones, a Nexus S, HTC sensation, Samsung Galaxy S2 and a Samsung Galaxy Nexus. One participant also installed the Appwhere prototype on his Nexus 7 tablet. All participants reported that they rarely arranged apps on their homescreen or uninstalled apps. Figure 5.18 displays screenshots of homescreens that were captured before adding the widget. Several homescreens have at least one duplicate app. P1 said that he "tr[ies] to avoid scrolling through the full list of installed apps on my phone as I have lots of them that I never use". P6 described his tablet homescreen layout as having a " a lot of space so I don't really worry about the layout. Kind of like my PC desktop". Figure 5.19 shows a screenshot of the Nexus 7 tablet.

Prior to the experiment, participants imagined ways that their app use might relate to their places. P2 predicted that he would “use more apps like gmail and google books in my car and apps like Facebook and Messaging in my bedroom”. P1 would “never use the Camera app at home when I have access to my digital camera. There are other apps such as Guitar Tuner that I would only ever use in the living room. I’m more likely to use my browser in the bedroom or when away from home, when I don’t have access to my laptop”. P6 shared that “in the kitchen I use TuneIn Radio or Google Music when cooking or doing housework. In the living room I’ll use Twitter/Facebook/Chrome while watching TV through my computer (Netflix)”. Other participants related their app use to connectivity, for example P5 is “more likely to use mobile Internet in the car and wifi in the house”, and P4 “can only use WhatsApp were there is a free wifi facility, outside my home address”. P1 expected that “the biggest influence on the different apps I use is whether or not I have Wi-Fi. I don’t tend use Facebook or Feedly on data. I might use Gmail or Google Maps when I don’t have Wi-Fi if I feel I have to, but wouldn’t do so if I could avoid it”. P3 did not believe that his app use related to places, “I usually use my phone when I am not at a PC and want to pass some time, I dont usually use it for ‘real’ work”.

### Research Methods

Participants were asked to position a Bluetooth beacon in five personal places where they used their smartphone most, and where they had sufficient ownership to leave their belongings. This requirement ensured that each participant had permission to secure the Bluetooth beacons for the entire experiment, and that the data tracked in places was most representative of where participants spent their time. All participants chose to put beacons in their Living room, Bedroom, Kitchen and Car. Some participants with offices also chose this as a place, and one participant had a large hallway that they chose as one of their places. Two participants could only think of four personal places to detect and did not use all of the beacons. All participants were familiar with each other, and shared at least one place with another participant during the experiment.

Following the process of [138], the widget was placed on the homescreen after 2 weeks, when enough historical data was collected to adapt the homescreen menu. The ranking of apps in the widget varied between participants, and participants were assigned to one of the following: most recently used, most frequently used, and most frequently used next (i.e. sequentially used).

At the end of the 4 weeks, participants were asked to fill out a final questionnaire about their experience. At this point, the beacons were gathered from participants, who were helped with uninstalling the experiment applications from their personal devices. Participant data was loaded into the visualisation tool after retrieving the Appwhere database from each device at

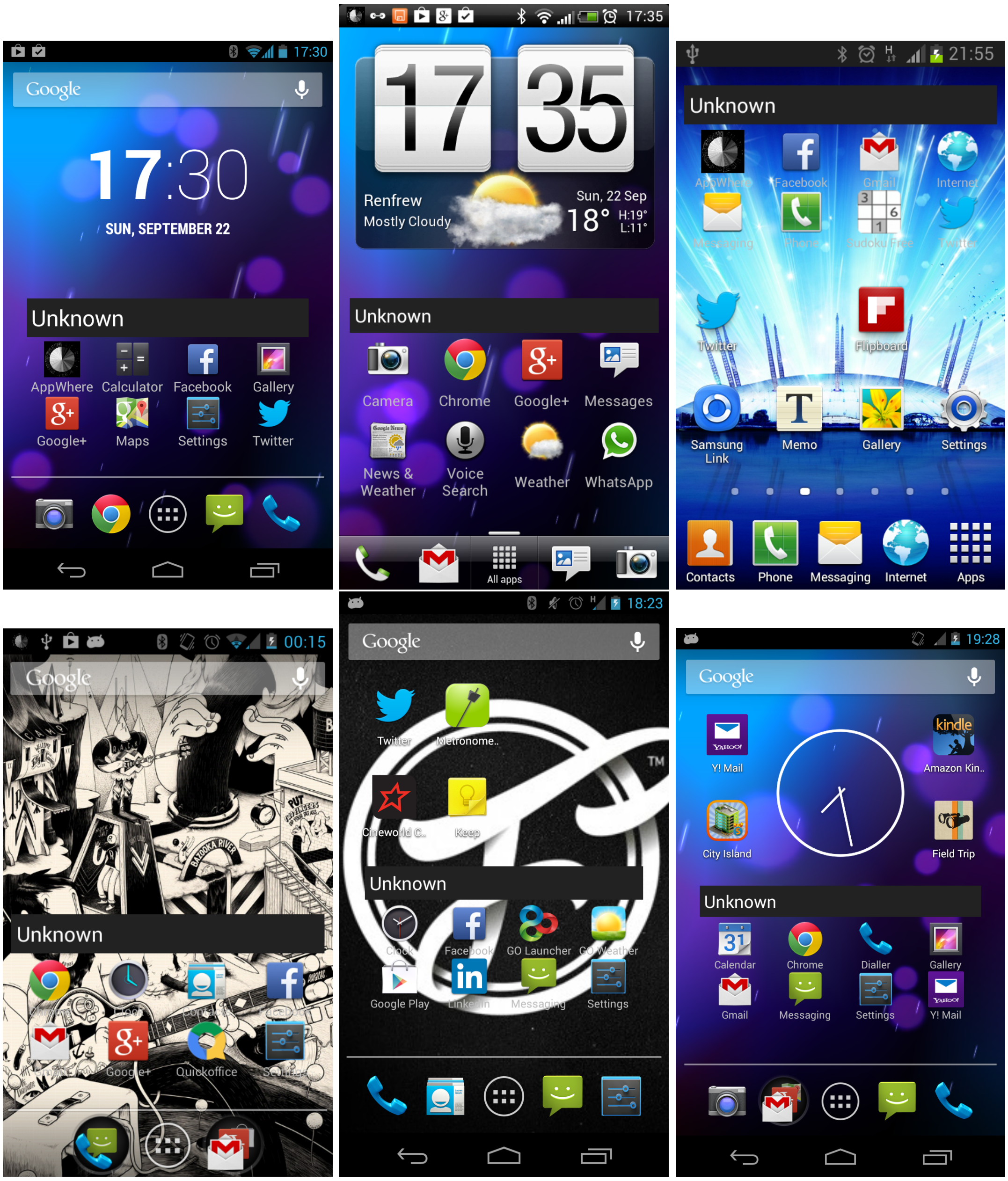
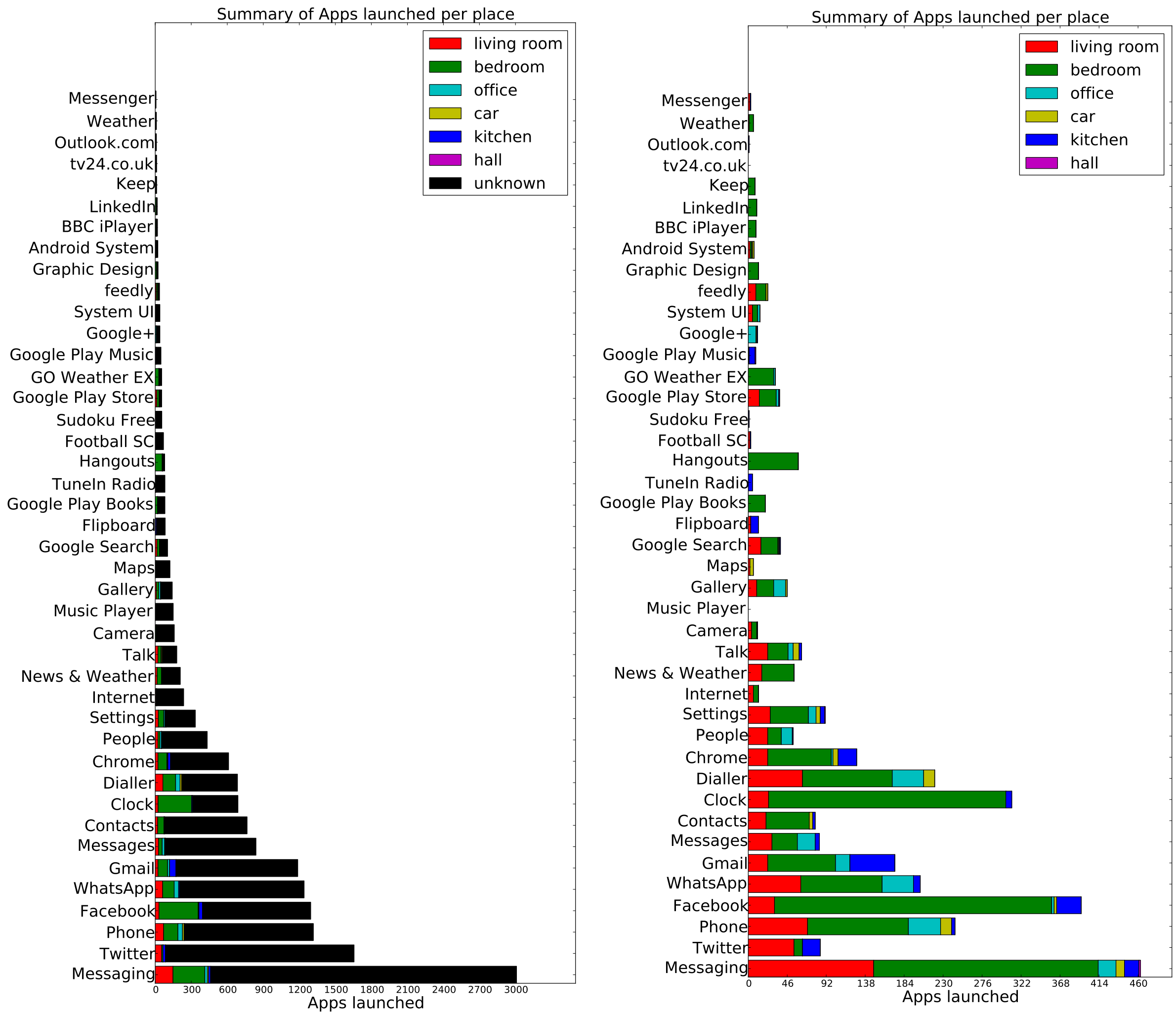



Figure 5.20: Screenshots of participant homescreens after installing the Appwhere widget. Screenshots were taken at the end of the study, after beacons were collected, and the Appwhere widget is reported to be in an 'Unknown' place.

the end of the study.

### 5.4.3 Results

The following sections discuss the data collected on app usage, the adaptive homescreen and information needs in places.




(a) Everywhere. (b) Personal places.

Figure 5.21: Summary of all participant app launches combined, with all data (Left) and data in places (Right). The order of apps are maintained to emphasise the difference in how frequently apps are used everywhere compared to in places. For example, Facebook is more popular in places than Twitter.

## App Usage

32,420 app launches and 11,591 screen off events were tracked for 6 participants over a period of 4 weeks. 6481 app launches and 2366 screen off events were detected in places, accounting for 20% of app launches and 20% of smartphone sessions. An average of 161 apps (SD=46) were installed on the devices, and 33 (SD=8) were actively used over the 4 weeks.

16.2 mins (SD=5.6) per day was spent in apps, and 8.7 mins (SD=6.3) on the homescreen. An interesting insight was that, prior to installing the widget, average homescreen visits lasted 28s (SD=24.7) before an app was selected, or 8s (SD=3.3) before turning the screen off. Afterwards, average visits lasted only 15.4s (SD=13.8), or 8.2s (SD=4.5) before the screen off event. This shows that the widget has the potential to reduce selection time, an average of 12.6s in this case. However, the time to select an app from the homescreen can be influenced by interacting with widgets or by external distractions, among other reasons. This means that it is not possible to verify timings on the homescreen without a controlled experiment, which was conducted separately in Section 6.1.

### Adaptive Homescreen

Screenshots of the adaptive homescreen widget on participant devices are displayed in Figure 5.20. In the second two weeks, 6558 apps were launched excluding system apps. The widget was used to launch 726 apps, 11% of all app launches. The widget was used most on the tablet, for 189 app launches. This was interesting, as the tablet was used to launch fewer distinct apps than was launched on the participant's smartphone. This suggests that the tablet is used for a more focused range of tasks. This insight also suggests that an adaptive widget could help to support tablet users with organising apps on the homescreen, which can display more menu items than the small smartphone screen. In addition to benefits for small screen sizes [50], this insight suggests that adaptive menus could still be valuable for larger mobile screens. Further investigation into form factors will be interesting for future work. The data collected provides a starting point to investigate the different use cases for smartphones and tablets. Participants found the widget "to be very helpful' [P1] and they "want[ed] to keep this widget as I like it" [P2]. P1 "found it convenient to launch apps from the widget and rarely needed to navigate to other pages of my home screen or browse my full list of apps. I also found it quite satisfying being able to see that the widget knew where I was and was attempting to order my apps accordingly - features that I don't want to give up!". P2 identified an issue with launching apps from the widget: "Sometimes when I tried to launch an app from the widget it would not launch, not sure why... besides from that I have no complaints". This defect was due to the implementation of the widget, as the shortcut icon responded to touch events instead of the full grid location.

### Information Needs in Places

Figure 5.21 (a) displays a summary of app launches of all participants. Some apps are launched more than others, and the long-tail distribution is clear. Colours in Figure 5.21 indicate different places. By tracking app use in places, more insight can be gained into information needs. Figure 5.21 (b) compares the app launches everywhere to app launches in personal places. Twitter was most used in the Living room, and Maps was mostly used in the car. Interestingly, the Clock is the 3rd most launched app in places, and only ranked 9th everywhere.

The interactive visualisation was used to explore app launch behaviour in places. Of the 8,843 app launch events in places, 54% were accounted for by P2, which is likely due to the small sample size of the experiment. 94.7% of his app launches in places were in the Bedroom, compared to 3.8% in the Office, 0.8% in the Living room, 0.6% in the car and 0.1% in the Kitchen. The remaining app launches in places were by P1 (16%), P4 (14%), P6 (7%), P3 (5%), and P5 (4%). Participants were most likely to be detected in their places in the morning, between 7am - 9am. Ignoring system apps, 594 apps were launched at this time, and the Clock accounted for 44% of app launches in this period. Some apps were detected to be used more in places than others. For example, P2 launched Facebook 294 times in the Bedroom, accounting for 57% of the times that he launched this app. This matches the expectations of P2 at the start of the experiment. Similarly, P1 did not launch Chrome in his Office, where he had his laptop. Devices also have the potential to be used differently in places. In the Living room, P6 launched 292 apps, and 60% were on his smartphone. Of the 217 apps that P6 launched in his Kitchen, 74% were on his tablet. His smartphone was used to launch the Messaging app in the Kitchen, which is a service that the tablet could not provide, as it did not have a sim-card. This suggests that P6 used his tablet more in the Kitchen, but required the Messaging app on his smartphone to send SMS. This highlights an interesting opportunity to share services between a user's devices, to avoid the user experience being sharded.

App launch activity was detected at all hours of day, and could be an indicator of sleep patterns. In the early morning, between 2am - 5am, 850 events were detected, and 43% were accounted for by P4. P4 was most likely to be detected in the Bedroom or the Living room at this period. Of the 219 app launches that were not system apps, the most frequently used apps were News & Weather (32%) and Chrome (24%). Chrome was most likely in the Bedroom, and News & Weather in the Living room. Apps can be explored that are more likely to be launched in places at certain times of day. For example, P1 launched Feedly, an e-reader, 36 times. In the morning, Feedly was more likely to be launched in the Bedroom, and the Living room was more likely in the evening. Though places correlated with time of day, this was not always the case. Of the 128 apps that P1 launched in the evenings between 5pm - 8pm, 43% were in the living room, 23% in the kitchen, 16% in the car, 13% in the bedroom, and 4% in the office. Furthermore, app launches in each place at this time were different: Though the top apps were all communication services, Google Talk was used in the car, the Phone in the Living room and Bedroom, Gmail in the Kitchen and the SMS in the Office.

The interactive visualisation can also be used to understand social information needs in places. For example, filtering by app name can show occasions where multiple participants are using the Phone and Dialler apps at the same time. Phone calls were initiated with the Dialler app, and P2 launched this app most in his Bedroom. By looking at app launches at

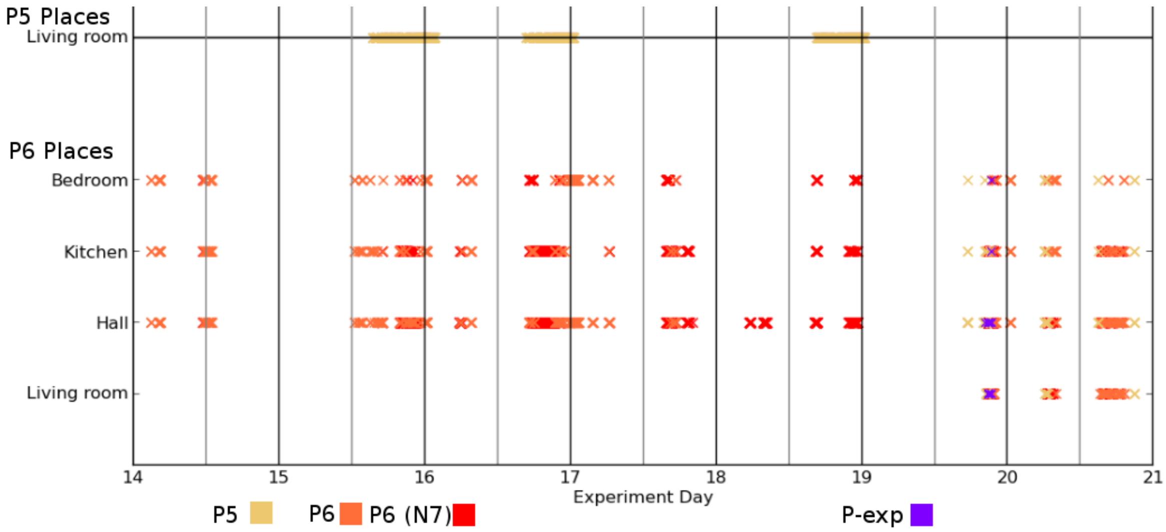



Figure 5.22: App use can be explored around other people. P5 (yellow) shared places with P6 (red/orange) while moving house during the experiment. The experimenter (P-exp, purple) is also detected in the place of P6 on day 19.

a date and time with similar durations, it could be possible to identify phone conversations between people. It is also possible to compare app use by multiple people who share a place. During the study, P5 shared places with P6 as she was moving house. As the beacon identifiers were hard-coded in this version of My Places, P6 was unable to add the beacons of P5, and so app use was not tracked in these places. However, this event is visible in the Bluetooth traces. Figure 5.22 shows the Bluetooth scans from P5, P6 and the experimenter. On days 19 and 20, P5 stopped detecting her own Living room, and detected the Living room of P6 instead. The experimenter was also detected for a short period on day 19. This demonstrates that Bluetooth can be used to find insights into app use around other people. For example, while staying with P6, P5 launched 569 apps, 46% more than her app use the week before. In comparison, P6 launched 2,012 apps, 34% less than the week before. It will be interesting to consider the social impact of app launch behaviour.

### Place Detection

Some participants experienced difficulty with the Bluetooth beacons. One challenge was that the beacons needed to be plugged in and turned on in order to be discoverable. If the beacons lost power for any reason, for example when the car engine was turned off, then participants had to remember to turn them on again. This issue can be overcome with the new LE beacons that were released after this study. Participants with older smartphones experienced difficulty with using Bluetooth along with Wi-Fi, for example, P5 found that 'the bluetooth interfering with the internet connection was a problem and I sometimes ended up turning it off as it made the device almost unusable at times'. This was a known issue[4] with some Android devices that will be fixed in newer hardware.

### 5.4.4 Discussion

The approach of tracking app use in personal places was demonstrated by marking places with traditional Bluetooth beacons. This contextual data was used to adapt the homescreen, and help mobile users organise their apps around the places where they are used.

An interactive visualisation was developed and was shown to provide insights into app launch behaviours in places. A limitation of the interactive visualisation was that places were presented independently, such that app use in the Living room of one participant was separate from the app use in the living room of another. A more flexible approach would be to only track the label assigned to the beacons, and allow participants to decide whether to track app use in each place differently or the same. Another limitation was that data could only be imported at the end of the experiment. It would be useful for the experimenter to be able to monitor app use as the experiment proceeds. This requires the app tracker to store data on a remote server.

Future work will include extending this visualisation to compare data from other sensors, including GPS and Wi-Fi. A more reliable form of indoor positioning will be required to perform long-term app tracking in places. The new LE beacons should be explored to improve the place detection and increase the reliability of app tracking in places. Users should also be given control to manage their places. Despite traditional Bluetooth beacons being an unreliable approach to place detection, it was demonstrated that a rapid prototyping approach allows interaction design that considers the opportunities of app tracking in places.

## 5.5 Summary

In this chapter, mobile interactions were explored that situate interaction around artefacts of personal places (RQ-1 of Section 3.1). Opportunities to situate mobile interaction around artefacts of personal places were extracted from responses to a questionnaire, and four types of artefact were identified that a smartphone user can interact with: objects, structures, people and places.

Using the Microsoft Kinect depth sensor, prototypes were augmented to become place-aware: web browser that enables websites to be bookmarked by tagging physical objects, and a digital book application that can be explored by moving around a physical bookshelf.

[4]https://code.google.com/p/android/issues/detail?id=41631

Feedback from evaluations with both of these prototypes show that interactions are made visible through movement, when they would otherwise have been private. The physical effort required to interact with the user environment may be excessive, compared to minimal interactions with a touchscreen display. Therefore, movement should be considered as a supplementary form of interaction. Participants also identified personalisation as an important factor of situating interactions in places. There could be more value in situating an interface around personal artefacts, than was found in these lab evaluations.

Bluetooth beacons enabled the development of Appwhere, a homescreen menu that adapts to personal places. A preliminary user study with this prototype was conducted, and despite the traditional Bluetooth beacons being too unreliable for detecting the context of a place, they were suitable for demonstrating the novel approach to app tracking in places. A dataset of app launch history was gathered, and an interactive visualisation was shown to be promising to gain insights from app use in places. Opportunities to build on were discovered, including improvements to the user experience of the adaptive homescreen menu, and the potential to track the social impact of app use.

The next chapter will iterate on design of Appwhere, with a focus on evaluating the stability of an adaptive homescreen menu, tracking app use in places with Bluetooth LE beacons, and a data-driven approach to evaluating the user experience of app navigation menus.

# Chapter 6

# App Navigation with an Adaptive Homescreen

The previous chapter considered ways of situating interaction with a smartphone around artefacts of personal places. This chapter focuses on the personal places where the smartphone is used, and how a place-aware adaptive homescreen might improve app navigation in places, and increase awareness of where apps are used (RQ-2 of Section 3.1). An adaptive homescreen menu is considered as a way to increase the accuracy of the selection of apps on the mobile homescreen, and an evaluation of the impact of stability on navigation time is performed. Insights gained from this experiment are used to update the Appwhere prototype from the previous chapter, with a focus on personal places and awareness of app launch habits. The design of this prototype is described, along with the results of a user study that improves the place detection of My Places with Bluetooth LE beacons, and leads to quantitative insights into app use in places. Choice in app navigation is briefly discussed, and the proposal of an objective model to evaluate the cost of navigation menus.

## 6.1 Stability in an Adaptive Homescreen

With access to a user's app launch history, a system can be developed that predicts apps that are most likely to be launched next. These apps can adapt in a menu automatically to provide fast access to a selection of apps. However, adaptive menus can become unstable, and changes can be frustrating and difficult to follow. [138] provide a detailed overview of features relating to mobile app use. The selection of apps is chosen by the adaptive model, which can be trained on a variety of features. Some features can be related to app use, such as frequency of use, and others may be contextual (e.g., time of day or location).

The homescreen is the main menu that is displayed on a smartphone. On Android, a small


Figure 6.1: The layout icons update when the yellow app is launched. The model and order of the layout have a different impact on stability.

number of apps can be placed on the homescreen, and the app drawer displays the entire list of apps. As a user installs more apps, the time and effort required to locate apps that do not feature on the homescreen will increase. Though [47] found that only a small number of installed apps are used frequently, the set that are frequently used changes over time [138]. Therefore, the homescreen needs to be organised regularly. However, arranging icons on the homescreen can be annoying and time consuming, and some users do not arrange their icons at all [18].

An adaptive menu can replace a section of the homescreen to support the user with organising the homescreen, and improve the usability of mobile devices. Stability on the homescreen is investigated as the adaptive model becomes less easy to understand. Two adaptive models are compared (most recently used, most frequently used next) and stability is controlled by the order of the layout (alphabetical, rank ordered). A usability experiment with 12 participants is performed to evaluate whether apps in an adaptive homescreen menu should be ordered alphabetically or by rank to improve the user experience of an adaptive menu. Research methods, including new stability measures for measuring stability in an adaptive homescreen menu, are discussed along with the results.

### Adaptive Menus

Menu design is an established field in HCI. [135] created Split Menus for desktop applications, and demonstrated that selection time can be decreased by moving or copying the top 4 frequently used apps to the top of the menu. [49] compared static, adaptive and adaptable split menus, with the naturally generated data of a single MS Office user. In comparison to an adaptive menu, which automatically adjusts to user behaviour, an adaptable menu is manually customised by the user, and a static menu does not update. The majority of their participants wanted a personalised menu, and were better at customising the adaptable interface after they had used the adaptive one. As the homescreen is an adaptable interface, this suggests that an adaptive component could support users with organising their app icons.

[57] note that adaptations can be more appropriate for novice users. [50] compared the impact of screen size in adaptive user interfaces, and found that an 'adaptive interface is more beneficial when screen real estate is constrained' and that 'adaptive interfaces are low risk for small screens' (p. 1254). This provides motivation for exploring adaptive menus on a mobile device.

*Accuracy* and *predictability* are two conflicting factors that affect the design of an adaptive menu. [59] define these as follows: 'a model's accuracy is the percentage of time that the necessary UI elements are contained in the adaptive area', and 'a model is predictable if it follows a strategy users can easily model in their heads' (p. 1271). While accuracy is controlled by the model, predictability depends on how a user perceives the adaptations. Predictability can be influenced by ease of understanding, frequency of adaptations, and stability.

Accuracy and predictability have been explored by [59], with an adaptive split interface that copies MS Office functions into a designated adaptive toolbar. The content of the adaptive toolbar and the sequence of button presses were predetermined to ensure the desired level of accuracy (50% and 70%), and predictability was controlled by comparing a most recently used model with a random model. It was found that increased accuracy improved selection time more than increased predictability. However, the most recently used model significantly increased subjective ratings, including control, predictability and the extent to which participants knew an item was in the toolbar. [58] found that accuracy can also increase the utility of the adaptive interface. In this experiment, models are considered that have high accuracy (approximately 85%).

Predictability can also depend on the frequency of adaptations, i.e. the rate at which updates to the user interface are applied. The impact of the frequency of adaptations can be illustrated by the results of [135] and [58], which show that increasing updates to the user interface from a slow pace (once per session) to a faster pace (up to once per interaction) decreases performance with the same interface. [58] discuss this result and 'suspect that the cause stems from the fact that high frequency effectively reduces a mechanism's predictability'. Holding back on updates to an adaptive menu will preserve stability during periods of interaction. However, limiting updates to the UI will reduce accuracy, since the state of the model will only be reflected in the UI at certain points in time. In this experiment, apps update after each interaction, and explore the impact of stability when updates are frequent.

### Stability

In this experiment, the impact of predictability and stability are considered in an adaptive homescreen. Stability has yet to be investigated on the homescreen. [167] considered stability in an app drawer with their Nihao Launcher, and found that 'a larger UI difference does not necessarily require longer app lookup time'. However, the app drawer contains all apps installed on a smartphone, an average of 177 apps reported by [138]. In comparison, mobile users develop an accurate mental model of the selection of icons on the homescreen, making it possible to create a usable Imaginary Interface that relies on this memory alone [64]. In the evaluation of the dynamic homescreen proposed by [138], 'participants stated that apps sometimes appeared and disappeared unexpectedly', and suggest that 'an effort can be made to minimize the movement of icons that persist between predictions, in order to reduce user distraction' (p. 181). Stability could help to improve the usability of an adaptive homescreen by making updates to the menu easier to follow and helping users to maintain their mental model.

Stability can be enforced by the adaptive model, as demonstrated in AccessRank [52] and [82]. However, as stability acts as a feature within the model, it increases the model complexity. Another approach is to use visual highlighting or shrinking [151], or to animate items gradually fading in [51]. To be spatially consistent, these techniques require all items to be displayed at once, which would make icons very small on a mobile device. [138] explored highlighting on the adaptive homescreen, by indicating the app with the largest increase in probability. This was found to be confusing for participants, especially when the highlighting was inaccurate. [133] suggest a simpler approach to stability: 'in situations where content changes slowly, the user would gain the benefits of developing spatial memory; in situations where items change frequently, the user could switch to an alphabetic arrangement (or a list view)' (p. 3147). Ordering apps alphabetically is easy to understand, and helps to increase stability by updating the list only when an item is inserted or removed. This technique is explored in the experiment.

If items are inserted and removed very frequently, then alphabetical order could be a less predictable strategy than displaying the most likely item at the top of the list, i.e. rank order. Therefore,the benefits of stability on the homescreen must be evaluated to determine its value as the adaptive model becomes less easy to understand. This will help interaction designers decide when stability should be included in the design of an adaptive homescreen.

### Adaptive Homescreen

The homescreen is the most common way of navigating apps on a smartphone, as found by [66] in their study of app launching habits. [16] found that users spend an average of 59.23 minutes per day on their device, with app use spread intermittently throughout the day. This presents a different use case to a desktop application, where the system is used in concentrated periods. Furthermore, apps can be installed and uninstalled on a smartphone, changing the range of functions that can be displayed over time [138]. Mobile devices are used in a variety of contexts, and this also affects the apps that are likely to be used [16].

To make accurate predictions, the adaptive model must update frequently to keep up with the continuously changing context. Therefore, it is important to consider the design of an adaptive menu in a mobile context, and understand how frequent adaptations will affect usability.

#### Research Hypotheses

The impact of stability is investigated as the adaptive model becomes less easy to understand. The adaptive model controls the movement of apps in the adaptive layout, and stability is controlled by varying the order: alphabetical order (alph) and ordering by the rank of the model (rank). The research question this experiment asks is: When items are added and removed very frequently, should stability be increased by displaying apps in alphabetical order? The hypotheses are as follows:

1. Rank order will decrease the perceived predictability.
2. Alphabetical order will improve selection time.
3. Alphabetical order will be preferred overall.

### 6.1.1 Research Methods: Measuring Stability of an Adaptive Homescreen

There are many factors that could affect how a user will perceive stability on the homescreen. For example, one person might perceive an interface to be less stable when the target app moves, whereas another might find it more noticeable when movements surround the target app. This makes it difficult to quantify stability in a single measurement. A variety of measures are adopted to compare the stability of the interface to account for this. This section defines the measures that are used to control stability in the experiment.

#### Similarity Measurements

List comparison measurements differ in two ways: weightedness and conjointness. In the ranked list of 8 apps, items can be inserted or removed, and so non-conjoint measures are required. Weightedness depends on the user: if she is more likely to look at the start of the list then a weighted measure should be considered, or a non-weighted measure if all positions are equally likely. A weighted measure would be appropriate when ordering by rank, and non-weighted when ordering alphabetically.

Learnability [31] is a stability measurement between 0 and 1 that considers how possible it is to learn the position of items in a menu. This is calculated as 'one minus the average distance that items move as a proportion of half of the total menu length' (p. 629). A stable interface would have a value of 1, whereas an interface that moves items randomly would have a value of 0. Learnability is used to compare the overall stability of the experiment conditions.

The distance between two lists can also be compared using similarity measures. In their review of similarity measures, [158] (p. 20) identify Average Overlap (AO) as a weighted non-conjoint measure, that applies weight to higher ranked items by averaging over matches in list prefixes of length $1 - k$. Kendall's tau is a non-weighted measure, and [46] (pp. 31 - 32) describe a way to transform this into a non-conjoint measurement, by considering a penalty for the case where items exist in one list but not the other. This penalty can be set to 0 to find the minimum distance, K (min), or 0.5 to find the average distance, K (avg). AO and K (min) are used to compare the similarity between adaptations in the experiment.

It is desirable to also measure the magnitude of movements in the list. [167] (p. 309) define UI difference to be the number of positions apps change between launches. This is a conjoint measurement, as it considers all apps in the app drawer. However, a non-conjoint measure is required that accounts for items that are inserted or deleted. One option is to consider items that do not appear in the top-k to have moved to position $k + 1$, an approach taken by [46]. Another approach is to assign a penalty of half the total menu length, as used by [52] in their calculation of Learnability. Rather than assigning a large penalty to insertions or deletions, a penalty of 1 is selected. This represents the appearance or disappearance of an item, rather than its movement from the end or middle of the list.

### UI Displacement

UI displacement is defined as a non-conjoint measure that assigns items that are inserted or removed to have a penalty of 1. This is calculated to be the mean sum of changes to the position of apps, where a change is:

$$\begin{cases} |({}_i - {}_j)| & \text{if position changed between i, j,} \\ 1 & \text{if the app is inserted or removed,} \\ 0 & \text{if the app position did not change.} \end{cases} \tag{6.1}$$

Both weighted and non-weighted UI displacement measures are used to compare lists, to account for both orders: alphabetical and rank. The displacement magnitude (DM) measures the total movement of items, and the weighted displacement magnitude (WDM) increases this value as movements occur towards the start of the list, where they might be more noticeable. WDM follows the approach of AO, but altered to account for the size of displacement

| Measure | **Example:** Dialler launched | Result |
|---|---|---|
| K (min) | $(1+1)/(0.5*8*7)$ | 0.07 |
| AO | $(0/1+1/2+3/3+4/4+...+8/8)/8$ | 0.8125 |
| Learnability | $1-(4/8)/4$ | 0.875 |
| TDM | $\|2-0\|$ | 2 |
| DM | $\|2-0\|+\|0-1\|+\|1-2\|+(0*5)$ | 4 |
| WDM | $(\|2-0\|)/1+...+(\|2-0\|+\|0-1\|+\|1-2\|+(0*5))/8$ | 8.371 |

Table 6.1: An example calculation is demonstrated by launching an app shown in the adaptive homescreen widget in Figure 6.2 (Right), which is considered to be using a most recently used model and apps are ordered by rank. If the Dialler is launched at index 2, three apps change position: Dialler moves up 2 positions to index 0, resulting in a TDM of 2. Chrome and Clock then shuffle down one position each, resulting in a total DM of 4. These movements are weighted towards the start of the list which is more noticeable with rank order, and so the WDM is relatively high.

in the UI. In an experiment environment, it is known what the next app to be selected will be, and so the target displacement magnitude (TDM) can be measured. These measurements are defined as follows, with $i_t$ = position of an app at time $t$, $k$ = the number of apps in the grid, and $n$ = the total number of app launches in the series:

$$\text{DM} = \frac{1}{n}\sum_{t=1}^{n}\sum_{i=1}^{k}|(\text{i}_t - \text{i}_{t-i})| \tag{6.2}$$

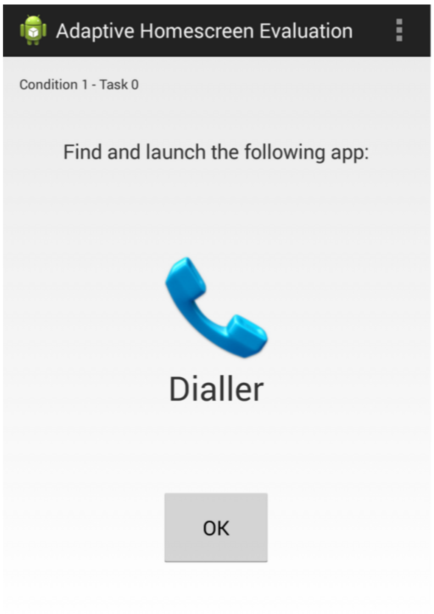
$$\text{WDM} = \frac{1}{n}\sum_{t=1}^{n}\sum_{i=1}^{k}\frac{1}{w}\sum_{w=1}^{i}|(\text{w}_t - \text{w}_{t-1})| \tag{6.3}$$

$$\text{TDM} = \frac{1}{n}\sum_{t=1}^{n}|(\text{target}_t - \text{target}_{t-1})| \tag{6.4}$$

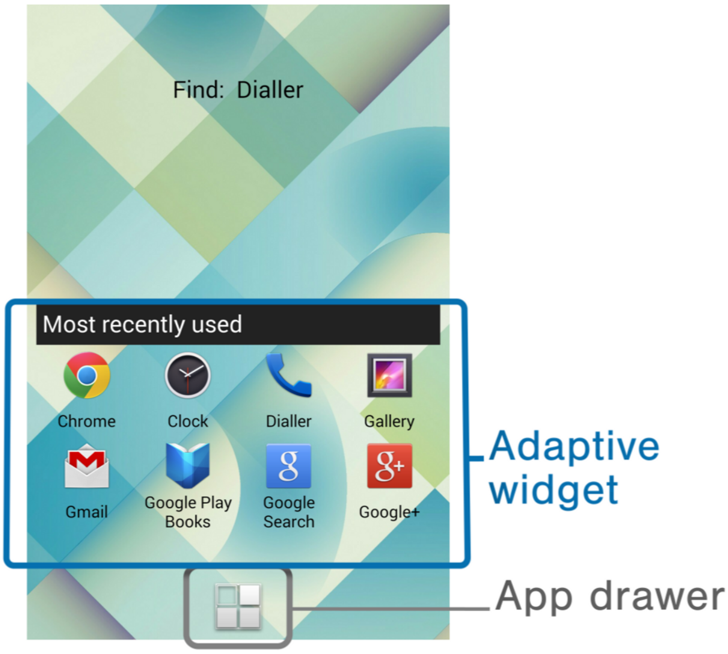
Table 6.1 illustrates the use of each of these measurements with an example. These measurements help to control stability in the conditions of the experiment.

### Dataset: App Launch History

To measure the time to select an app, it is required to record when a user starts to search for an icon and when the icon is selected. All distractions must be eliminated during the search. This requires a controlled experiment. However, it is important to consider how the adaptive homescreen will be used outside a research environment, with a user's own apps. To make the distribution of apps in the experiment dataset less contrived, a naturally generated dataset is used, the same approach as [49]. This dataset contains actual app history that was collected




(a) The experiment app.




(b) The custom homescreen, with the adaptive widget in the lower section. Above it, a separate widget displays a reminder of the app to select: in this case, 'Find: Dialler'.

Figure 6.2: The experiment app.

in Section 5.4. The history of one test subject was selected as the dataset of this experiment. Using the same dataset for all participants makes it possible to compare selection times for each experiment task. Additionally, the launch history of multiple test subjects were not combined into a dataset as this would disrupt the natural app cycles that may be important for the MFU-n model. The launch history of P2 was chosen as the dataset in this experiment, since his smartphone usage was most representative of the target user group for an adaptive homescreen: P2 used the most downloaded apps (32 apps), he had 147 apps installed on his device, and he launched an average of 146 apps per day (SD=65). His most common app launches belonged to the Built-in, Social and Communication categories. After system apps were eliminated, including the Launcher, and launches that were repeated more than 3 times, the app launch history dataset contained 1275 launches.

### 6.1.2 Research Design

A usability experiment was performed to investigate the impact of stability on the homescreen as ease of understanding decreases. The experiment was conducted in a controlled lab environment and lasted 30 minutes. A within-subjects design with four conditions was used: MRU and MFU-n, ordered alphabetically and by rank. These were counter-balanced using a Latin square. This section describes the design of the experiment, including the adaptive models, the adaptive layout, and the experiment task.

### Adaptive Models

With even a small number of features, it becomes difficult to understand the behaviour of an adaptive model. Two models are compared in this experiment, which control the filtering of apps in the adaptive menu. The models depend on features related to app use so that they can be controlled in a usability study, and they are accurate enough for short experimental conditions, as follows:

- *Most Recently Used (MRU)* considers the last time that each app in the history was launched.
- *Most Frequently Used Next (MFU-n)* counts the number of times that each app has been launched after the previously used one, i.e. sequentially used.

The Most Recently Used (MRU) model is considered to be easier to understand than Most Frequently Used Next (MFU-$n$), since it is easier to recall recently used apps than to keep track of apps that are commonly used after one other. In contrast, MFU-$n$ has the potential to be more accurate over a longer period of time as it reacts to two pieces of context: launch frequency and the previously used app. These models vary in ease of understanding, and change frequently enough to be noticed in a short lab experiment. However, it is not known how their UI displacements compare. To calculate the UI displacements, the dataset was used to simulate the adaptive homescreen.

A simulation of the dataset in the adaptive homescreen verified that MFU-$n$ caused more UI displacements and updated more frequently than MRU. It was found that MRU updates slowly over short-term use. In comparison, MFU-$n$ can display a completely different set of apps after each interaction. A sparsity issue was also found with MFU-$n$, resulting in an under-populated menu. This means that some apps are followed by fewer than 8 distinct apps in a single session, where a session is the period between turning the screen on and off again. This could be avoided by using a hybrid model: For example, the overall most frequently used apps could fill the remaining spaces. However, this might not be desirable, as fewer apps reduces the clutter of irrelevant apps on the homescreen, which could reduce selection time. In the experiment, periods of low menu population are allowed, and the impact of menu sparsity on usability is investigated.

### Adaptive Algorithm

The following process occurs each time an app is selected:

1. The model updates its ranking.

2. If the ranking changes, then apps may or may not change position depending on the order.
3. If any app changed position, then the layout is updated.

For example, the rank of MRU updates when a selected app differs to the previously used app. If the rank changed, then the alphabetical order will only update if the most recent app is not already in the layout. Any updates to the UI execute before the user returns to the homescreen.

### Adaptive Layout Design

The adaptive widget is designed as a grid layout, which is available for Android 4.0+ smartphones. This can be seen in relation to the app drawer icon in Figure 6.2 (Right). The label above the grid reveals the model that is in use: it states either the name of the model, or for MFU-$n$, a reminder of which app was used previously. Positions in the grid are indexed left-to-right, top-to-bottom, as one would read a page of English text. Though [167] found that users do not necessarily read grid items as a list, this is a common approach to grid indexing that is used in the app drawer and that smartphone users are most familiar with. Alternate approaches are possible, such as prioritising items at the edges or the middle.

The widget was placed in the lower section of the screen, where it is comfortable to access icons in case of one-handed interaction [18]. It also features above the app drawer, which minimises its distance to the full selection of apps, similar to a split menu [135]. Though split menus recommend a maximum of 4 items for users to scan quickly, 8 apps were chosen, as used in a split interface [58], to increase the accuracy of the simple models to 85%, and to create more opportunities to interact with the widget during the short lab conditions. As a model becomes more accurate, the need for a second row of items may become unnecessary. However, there could still be benefits to increasing the number of items, including to promote stability, or to outweigh the potential cost of entering the app drawer. The impact of varying the number of homescreen apps on accuracy is discussed in more detail in [138]. Though it is also possible to adapt folders or homescreen pages [18], this was not considered since they require additional interactions that increase selection time [66].

### Participants

The 12 participants were aged 21 - 40, all were smartphone users from a computing science background, and one was female. Participants had no prior experience with the Appwhere prototype.

### Equipment

The experiment app was installed on the Nexus 4 smartphone that was used by all participants. 150 apps filled 6 pages of the app drawer, which is close to the average reported in [138].

### Task: Gamification of App Launching

Participants were asked to perform tasks with an adaptive homescreen widget that changes the position of apps in a layout according to the chosen models and orderings. The task was designed as a game, an approach also demonstrated by [17]. There were 4 sets of 60 selections, using the widget or the app drawer as required. A custom homescreen was used to ensure that apps could only be selected from these two menus, which was implemented with the open source ADW Launcher.

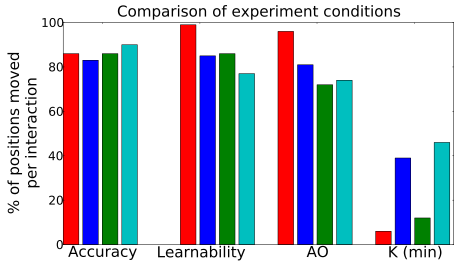
Participants were presented with a task UI that displayed the icon of the app that they should select next. Screenshots of the task UI and adaptive widget are shown in Figure 6.2. When a participant selected the correct app, the task UI launched to display the next app in the series. If a selection was incorrect, participants were immediately informed by a short notification near the bottom of the screen. To ensure that the widget did not update if an incorrect app was selected, adaptations were controlled using a predetermined list of widget states, which was compiled by simulating the widget with the app launch history dataset. Though incorrect selections do happen in real life, this control makes it possible to compare the results between participants.

As the participants were required to use another person's apps, the experiment method provides time to become familiar with the target app before each selection task. Between tasks, the 'OK' and hardware 'back' buttons were disabled for 2.5s to enforce a minimum pause. Participants could familiarise themselves with the icon of the next app to be selected before continuing with the selection task. The selection time for the previous task was also displayed during this period. This break replicates natural app use behaviour, since there is usually a period between clicking on an icon and returning to the homescreen where a user interacts with the app.

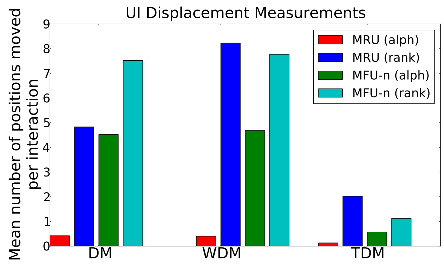
Participants were informed that the first 10 selections of each condition were for practice, which left (50*4*12) = 2400 selections to be analysed in the results. To motivate participants to select apps as quickly as possible, a prize was offered for the user with the lowest overall selection time.

| | Conditions | | | |
|---|---|---|---|---|
| | MRU (alph) | MRU (rank) | MFU-*n* (alph) | MFU-*n* (rank) |
| Accuracy | 0.86 | 0.83 | 0.86 | 0.9 |
| Target position | 4.82 | 0.02 | 4.10 | 0.88 |
| Grid population | 8 | 8 | 6.78 | 7.23 |
| K (min) | 0.06 | 0.39 | 0.12 | 0.46 |
| AO | 0.96 | 0.81 | 0.72 | 0.74 |
| Learnability | 0.99 | 0.85 | 0.86 | 0.77 |
| TDM | 0.13 | 2.02 | 0.57 | 1.12 |
| DM | 0.42 | 4.83 | 4.52 | 7.52 |
| WDM | 0.40 | 8.23 | 4.68 | 7.77 |

(a) Summary of experiment conditions, with the mean of measures. The mean target position is lower with rank order, and mean grid population is lower with MFU-n.




(b) Stability measurements.




(c) UI displacement measurements.

Figure 6.3: Comparison of stability measurements in each condition.

### 6.1.3 Statistical Design

This section explains how dependent variables were measured in the experiment, and how independent variables were controlled by selecting blocks of app launch data for each condition.

#### Task Dataset Selection

The task dataset comprises 4 blocks of 60 app launches, and each non-overlapping block is used in the selection task. Blocks were chosen from the second half of the dataset. The whole sequence of app launches leading to the start of each block was used to populate the model, and defined the starting state of the layout for each condition.

To evaluate the impact of stability, a noticeable difference was required in the average number of UI displacements between the conditions. Sequences of app launches were chosen that fit this criteria by running a simulation of the widget. To ensure that each condition had a high accuracy (approximately 85%), blocks of app launches were selected that had the required number of hits. Stability was controlled by selecting a difference in DM of at least 25% between alphabetical and rank order conditions. Independent variables are summarised in Table 6.3 and the comparison of stability in each condition is visualised in Figure 6.3.

Table 6.3 shows that the average target position in each condition is lower when items are ordered by rank. This is as expected, since the most relevant apps are more likely to be near the start of the list, i.e. position 0. The table also shows that the grid population is lower on average in the MFU-*n* conditions. This is because fewer than 8 distinct apps can be used after the previous one, as found during the model simulation.

In Figure 6.3, Learnability is greater than 0 in all conditions, which means that they are more stable than a random algorithm. Learnability is also higher in the alphabetical conditions compared to rank order, and the UI displacements are lower. Both of these these measurements indicate that an alphabetical order makes the models more stable, compared to rank order.

Both orders of MFU-*n* were noticed to have a similar AO. This means that a similar number of items change position near the start of the list, which might be surprising for alphabetical order. Changes towards the start of the list could increase if the layout is under-populated, and also if a completely different set of apps is displayed after each interaction. If one were to search from the start of the list, the stability of MFU-*n* could appear to be similar for both orders. However, the target position for MFU-*n* (alph) is likely to be somewhere in the middle of the list, and the apps surrounding the target will act as alphabetical landmarks that will influence search in this condition. Therefore, it is expected that K (min) will be more reliable than AO, and that MFU-*n* (alph) will be perceived to be more stable than MFU-*n* (rank).

### Task Measurements

For tasks where the app is in the widget, the selection time is recorded. This is measured as the time between clicking the OK or back button to clicking on the correct app icon. For tasks that require the use of the app drawer, the decision time of when it is opened is recorded. This denotes the time that a participant realises the app is not in the widget and that they have to use the app drawer. Errors are also recorded, which are any incorrect selections and extra swipes or clicks that are performed.

At the end of each condition, users were asked to rate their subjective opinions on a 7-point Likert scale (1=low, 7=high). After the experiment, participants were asked to rank the conditions in order of preference (1=high, 4=low). Both questionnaires are available in Appendix D. Subjective comments made by participants were also recorded.

### Statistical Methods

A 2 (alphabetical vs. rank order) x 2 (MRU vs. MFU-*n*) ANOVA was performed to test the significance of selection time, with $p < 0.05$. A log transformation was used to control for non-normal distributions in this timed data. A two-tailed Mann-Whitney U-test was used to test ordinal data, with $p < 0.05$ and $U \leq 37$ selected for the 12 participants. To compare the

| | | Conditions | | | | Averaged over ordering | | | Averaged over predictability | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | MRU (A) | MRU (R) | MFU-*n* (A) | MFU-*n* (R) | Alph. | Rank | Sig.? | MRU | MFU-*n* | Sig.? |
| Data | Accuracy | 0.86 | 0.83 | 0.86 | 0.9 | 0.86 | 0.87 | | 0.85 | 0.88 | |
| | Mag. change | 0.55 | 4.97 | 4.95 | 7.72 | 2.75 | 6.34 | * | 2.76 | 6.33 | * |
| Time | Duration (s) | 1.21 | 1.44 | 1.45 | 1.42 | 1.33 | 1.43 | * | 1.33 | 1.43 | * |
| | Decision (s) | 1.46 | 2.15 | 1.87 | 2.04 | 1.66 | 2.09 | | 1.80 | 1.95 | |
| Subjective | Awareness | 5.5 | 7.0 | 6.5 | 5.5 | 5.75 | 6.0 | | 6.0 | 6.0 | |
| | Predictable | 5.0 | 7.0 | 2.5 | 3.0 | 3.75 | 5.0 | * | 5.75 | 3.0 | * |
| | Useful | 6.0 | 6.0 | 4.5 | 5.0 | 4.5 | 5.25 | | 5.75 | 4.25 | * |
| | Satisfied | 5.0 | 6.0 | 3.5 | 4.0 | 4.25 | 4.75 | | 5.5 | 3.75 | * |
| | Efficient | 6.0 | 6.0 | 4.5 | 5.0 | 4.75 | 5.5 | | 6.0 | 4.25 | * |
| | Control | 5.0 | 6.0 | 2.0 | 3.0 | 3.5 | 4.5 | | 5.5 | 2.5 | * |
| | Frustrating | 2.0 | 2.0 | 2.5 | 3.0 | 2.5 | 2.5 | | 2.0 | 3.0 | * |
| | Overall Pref. | 1.0 | 2.0 | 4.0 | 3.0 | 2.5 | 2.5 | | 1.5 | 3.5 | * |

Table 6.2: Summary of results: mean of measured data, median of subjective data and mode of overall preference. The results of the two independent variables (order and ease of understanding) are isolated in the right side of the table by averaging between the two orders and models.

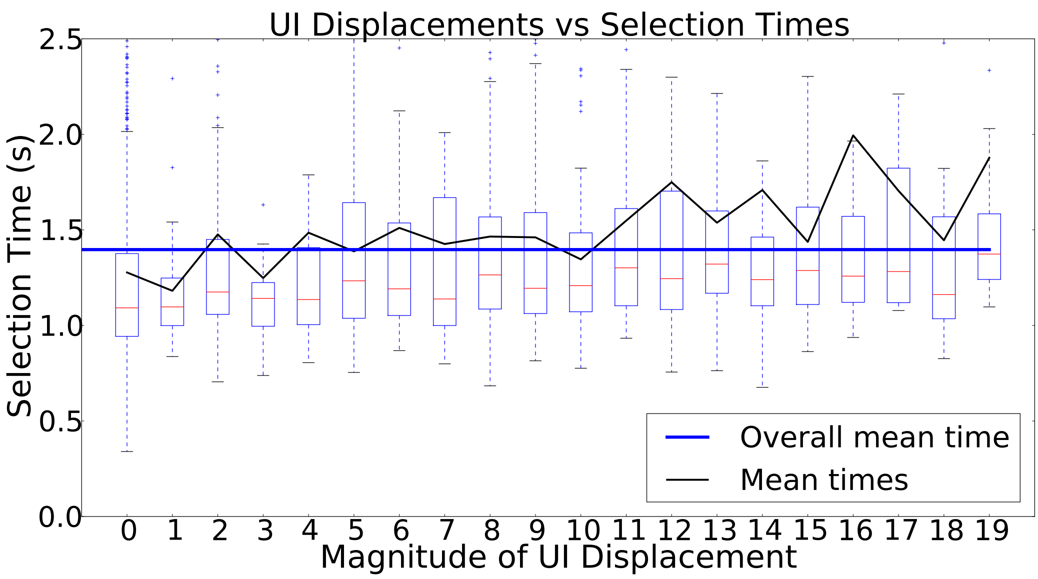
correlation between selection time and UI displacements, the Pearson coefficient was used, with Evan's [45] guide for interpreting the r value ($0.2 \leq r \leq 0.39$ for weak correlation).

### 6.1.4 Results

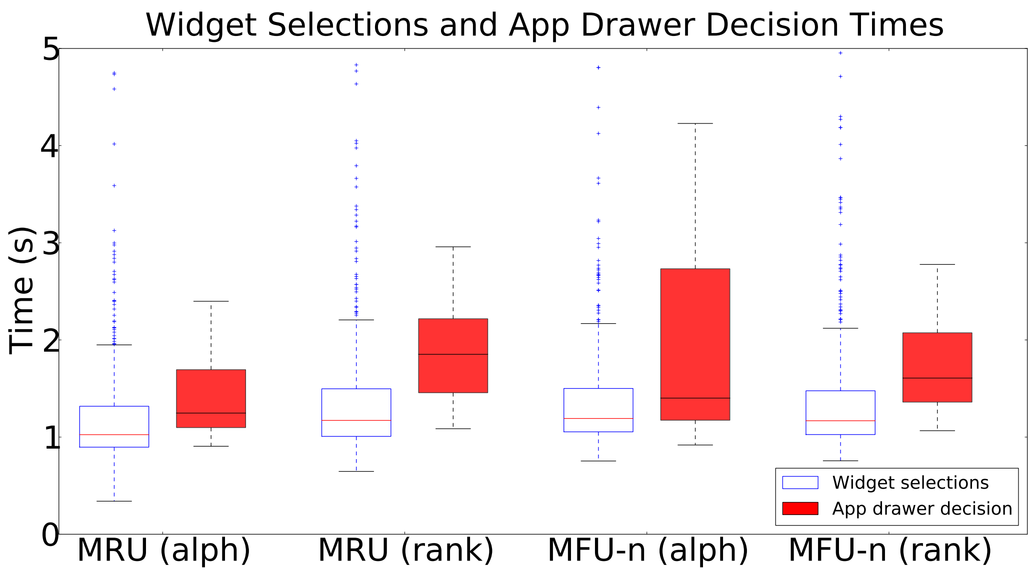
Table 6.2 provides a summary of the results. The number of errors made during the experiment were negligible and so are not reported. The selection time, predictability and overall preference are presented.

#### Selection times

The overall mean selection time was 2.08s (SD=2.7s), and 1.39s (SD=0.9s) using the widget alone. Figure 6.4 (a) compares the widget selection times to the UI displacements. It is clear that the mean selection time increases with the magnitude of the previous UI displacement. Selection time increases above the overall mean selection time when there are 6 or more UI displacements. The Pearson correlation for this effect is weakly significant [45] ($r=0.208$, $p<0.001$).



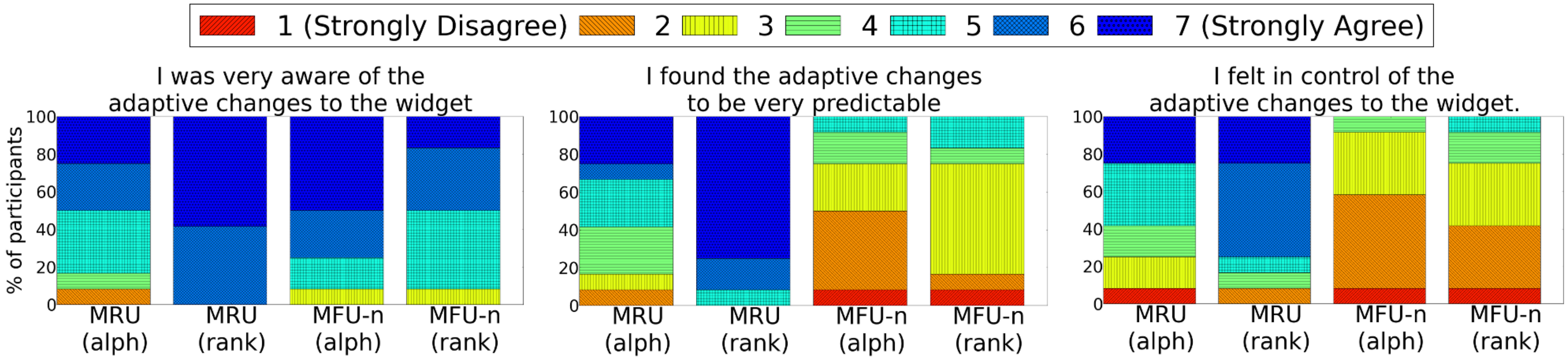

(a) Widget selection times increase with increasing numbers of UI displacements (DM), and is above the overall mean widget selection time (1.39s) when there are 6 or more UI displacements. (Note: this chart is capped at 2.5s to reduce the visual effect of outliers).




(b) Widget selection times (left) were significantly faster in MRU (alph) than the other conditions.

Figure 6.4: Selection results.

Figure 6.4 (b) displays Boxplots of the widget selection times and decision times. Widget selections were signicantly faster in the MRU (alph) condition compared to the others ($p<0.001$), with a mean selection time of 1.21s (SD=0.71s). Decision time was also faster (1.46s) in this condition, but a larger sample of selections from the app drawer would be needed to test if this result is signicant.




(a) Awareness. (b) Predictable. (c) Control.

Figure 6.5: Subjective ratings for the changes to the adaptive widget. In the MRU (rank) condition, participants were more aware of changes, and felt more in control and predictable.

### Predictability

The subjective ratings can be viewed in Table 6.2. As expected, both MRU conditions were rated more *predictable* than MFU-*n* (U=2.5, $p < 0.001$), and participants felt more in *control* when using this model (U=9.0, $p < 0.001$).

The MFU-*n* conditions were rated significantly lower in *satisfaction* (U=13.0, $p < 0.001$), *usefulness* (U=26.5, p=0.008), *efficiency* (U=30.5, p=0.015) and were more frustrating (U=34.5, p=0.029). Despite the MFU-*n* model being as accurate as MRU, participants were "not sure how it was adapting" and "didn't like that the homescreen changed unpredictably sometimes" [P9].

The perceived predictability significantly increased when apps were ordered by rank, particularly in the case of MRU (U=29.0, p=0.01), as displayed in Figure 6.5. This was mentioned in the feedback to the MRU (alph) condition, where one participant commented, "it was difficult to remember what the least recently used app was [and so] it was difficult to predict which would be dropped when a new app was opened" [P10]. Conversely, another participant noticed that MRU (rank) "seemed to sort the frequently used ones within reach of my right thumb" [P9]. Ordering MRU by rank did have its drawbacks, as some participants felt that it caused apps to "move from the end of the widget to the start [and] was quite annoying" [P8]. Additionally, P2 "could [only] predict the adaptive change for apps I had just used but could not remember the last 8 apps used".

The ratings for the *awareness* of the adaptations are displayed at the top of Figure 4. For MRU, participants were significantly more aware of updates when apps were ordered by rank (U=33.0, p=0.017), which is consistent with the measured number of UI displacements. In comparison with MFU-*n*, participants were more aware of movements when apps were ordered alphabetically, when the measured number of UI displacements was lower. Though this awareness was not significantly more (U=46.5, p=0.124), it is still an interesting result, as it is in conflict with prior expectations and it is consistent with the measured AO for MFU-*n*.

### Overall preference

Figure 6.6 displays the ratings for overall preference. In addition to being the fastest, participants preferred MRU (alph) overall, with 50% of participants rating this most preferable, and 25% rated this their second preference. This combination of model and ordering was the "best version of adaptive widget because [it was] very predictable" [P9].

At the lower end of the scale, MFU-*n* (alph) was both the slowest and the least preferred overall, rated least preferable by 50% of participants and third preferable by 25%. In this condition, participants said, "often I had seen the position of an app just before I was told to

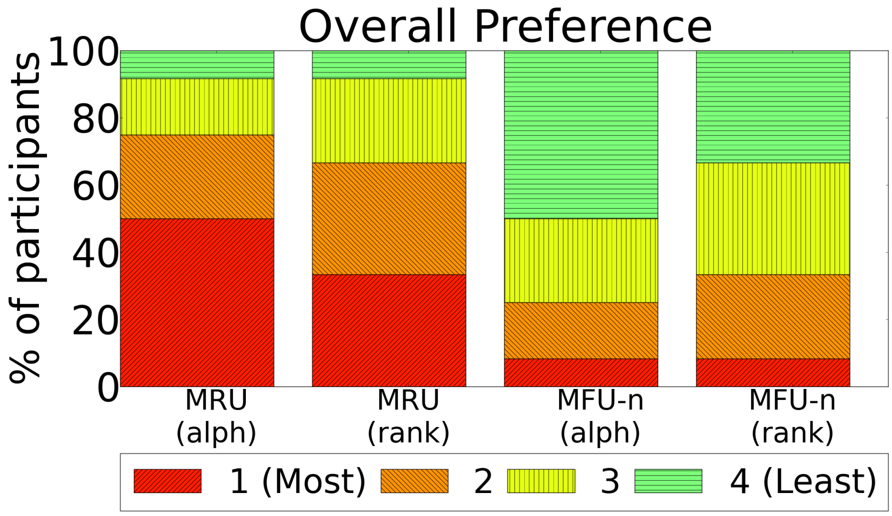



Figure 6.6: Overall preference as a percentage of participants.

press it, however it had then moved" [P2], and "the widget was not always fully populated and it felt as though major changes (> 2 apps) occurred frequently" [P10].

There was no significant preference between MRU (rank) and MFU-$n$ (rank). Although MRU (rank) was preferred more than the alphabetical order of MFU-$n$ (U=32.5, p=0.018), it was not preferred significantly more than MFU-$n$ with rank order (U=40, p=0.056).

### 6.1.5 Implications for Design

The results have several implications for the design of adaptive homescreens, which are highlighted in this section.

*Rank order is effective when updates are frequent.* Participants were faster and preferred when the MFU-$n$ model was ordered by rank, the condition that caused the most UI displacements. This result is contrary to the hypothesis, and the suggestion of [133]. Instead, it is important to support a search strategy that is easy to understand, even if this strategy does not increase stability. When items update frequently, participants found it easier to look towards the start of the list than to use alphabetical order. This result will inform the design of more reactive, adaptive homescreens. Further work will be required to test if this will generalise to other approaches of stability.

*Rank order increases predictability.* The subjective results show that predictability increased when both MRU and MFU-$n$ models were ordered by rank. This result was contrary to the first hypothesis, as it was expected that an alphabetical order would increase predictability by stabilising the interface, and allow items to be indexed by app name. However, rank order was more transparent about which app would be dropped when another was inserted. In comparison, alphabetical order did not reveal the relative value of items, and so items could unpredictably appear and disappear at any position. MFU-*n* was even less predictable when ordered alphabetically, since many items could appear and disappear at once, making the alphabetical landmarks very unstable.

*Large movements were irritating.* Though participants could understand how apps moved in the layout, some found it irritating when the magnitude of movement was large, particularly when items moved from the end of the menu to the start in MRU (rank), which has a DM of 14. Further work will be required to investigate this effect, particularly as users use their apps intermittently in the wild. A compromise might be to use a stable ranking algorithm, such as AccessRank [52], but allow items to move towards the start of the list while minimising large movements.

*Fewer grid items are easier to recall.* With MRU, participants were unable to recall all 8 recently used apps, and this limited their ability to know whether an app was still in the menu. If a user expects an app to be in the list when it was not, then this could impact selection time and subjective opinion. Fewer items in the menu could reduce this effect, such as the 4 items recommended for a Split Menu [135]. This number of menu items will impact the accuracy of the adaptive model, which should be considered in the design of the adaptive homescreen.

*Fewer grid items reduce decision time.* The time to decide that an app is not in the layout was faster in both MFU-*n* conditions compared to MRU (rank). This could be because the MFU-*n* model did not fully populate the layout, as fewer items to check would make this decision easier. The decision time could be faster with rank order if there were fewer items. However, fewer items will reduce the accuracy of the adaptive model. This suggests a trade-off, as an accurate model creates fewer occasions that an app is not in the grid. This trade-off should be considered when selecting the number of items on the homescreen.

*UI displacements increase selection time.* Selection time increased with the magnitude of the previous UI displacement, and was above average when there were around 6 or more displacements, which was close to the mean DM for MFU-*n* (rank). This was found to be weakly significant, and further work would be required to find this threshold. With an estimate of how frequently the interface can update before negatively affecting performance, practitioners will be able to decide whether alphabetical order will improve the usability of their adaptive menu.

*Alphabetical order is effective when updates are infrequent.* Participants preferred alphabetical order for the MRU model, which added and removed items very infrequently. This result is unsurprising since it is the version that is most commonly used for quick-launch menus. Additionally, decision time was lowest with MRU (alph), and highest with MRU (rank), which highlights the effectiveness of alphabetical order when a model updates items infrequently.

### 6.1.6 Discussion

An adaptive homescreen can help mobile users to make better use of their limited screen space. However, adaptations make the UI less stable and less easy to understand. The impact of stability on selection time and subjective rating was considered as items are added and removed frequently. UI displacement measurements were defined that helped to control stability in the usability experiment. It was demonstrated that the combination of the layout order and the dynamics of the model cause stability to decrease. Naturally generated data was used in define the selection tasks in the experiment. The UI displacement measurements were used to analyse this dataset, and to select blocks of app launches for the experiment conditions. The results show that when items update infrequently, participants were faster and preferred items ordered alphabetically. However, when updates were frequent, participants were faster and preferred rank order. This result will inform the design of more reactive adaptive homescreens.

The experiment only considered simple models of app launch history. As UIs become increasingly sensitive to context, there is a risk of providing less transparency for the user. This can be problematic, especially when there is a mismatch between the user and system in the understanding of context. Intelligibility in an adaptive system is important to allow the user to share an understanding of the system state. Negotiated interactions could provide a way for users to influence the state of the system when more control is desired. Future work in this area will also involve testing the impact of UI displacements on predictive models. As the results are limited to a controlled usability environment and a limited population of users, stability in the adaptive homescreen should be evaluated with more users, over a longer period of time, and with each user's personal app launch data.

## 6.2 Information Needs and Self-Reflection in Places

An adaptive homescreen can reduce time spent navigating apps, by automatically organising the homescreen with the apps that are most likely to be launched. Automating the annoying and time consuming task of organising the homescreen allows the user to reach the apps that they need quickly, and has the potential to reduce focus on a private display. A model that is easy to understand can improve the usability of an adaptive homescreen, as found in Section 6.1. Therefore, it is desirable to simplify the understanding of predictive models. In the physical world, complex concepts can be understood with logical labels. For example, a user who is at home can be predicted to be in a dining room during meal times. In comparison to adaptive models that adapt to time of day and geographic location, it could be easier to understand a model that reacts to places. Adapting to personal places has the potential to simplify the adaptive model, and improve the navigation of apps on the mobile homescreen.

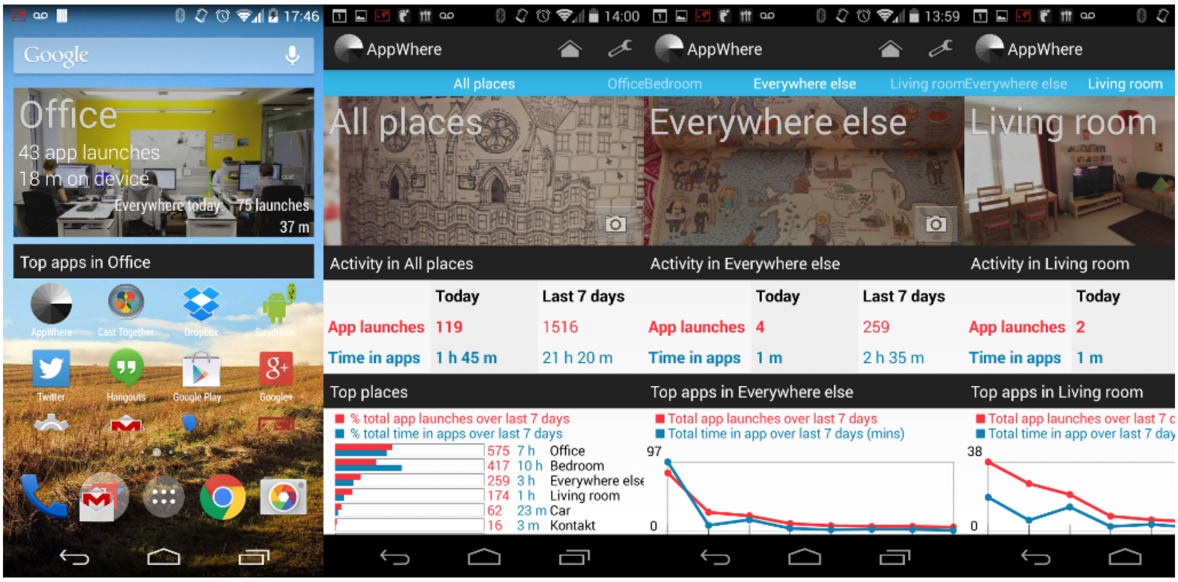



Figure 6.7: Adaptive homescreen prototype. Users can assign an image to each place. The Appwhere app displays stats on usage in each place.

Mobile information needs in places have most commonly been explored in diary studies [28, 30, 38, 141, 77, 153]. Automating the tracking ofapp use in places can reduce the burden from the user, and allows quantitative app use data to be gathered. In a study of the contextual usage of mobile smartphone services, [85] collected data from 140 Symbian smartphone users over 2009 and 2010. From user data produced for an average of 134 days, it was found that a majority of active smartphone use was in the home, which provides motivation for tracking app use in personal places. Self-reflection applications encourage users to quantify themselves by collecting long-term data [94, 125]. With a better understanding of ones own habits, it becomes easier to change behaviours. Persuasive applications can build on this data to motivate behavioural change, such as AppDetox [98] that encourages the user to disable addictive mobile apps.

Appwhere, a homescreen menu that adapts to personal places, was demonstrated with traditional Bluetooth beacons in Section 5.4. This prototype is improved with more robust place detection by using Low-Energy Bluetooth (LE) beacons. Awareness of app use in places is improved by providing more usage summaries, and making the app use statistics more visible on the homescreen. A user study is performed with 5 participants over a period of 5 months, and the insights gained from quantitative app use in personal places are discussed.

### 6.2.1 Prototype Design: Appwhere 2.0

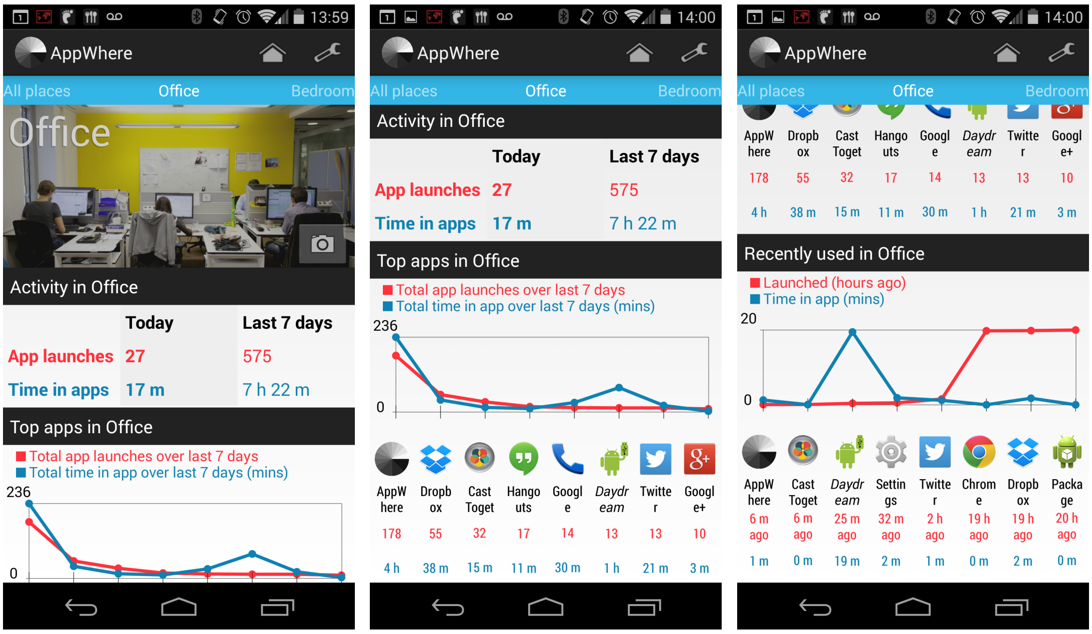
The Appwhere prototype consists of three parts: an app tracking service, adaptive widget and usage statistics. A focus on personal places is realised by presenting the apps that are most used where the user is located. Awareness of app use is provided as statistics on the


Figure 6.8: Kontakt.io Low Energy Bluetooth (LE) beacon placed in a car.

homescreen and as a visualisation in the Appwhere app. In addition to the design of Appwhere, place detection with Bluetooth LE beacons is improved, along with the interactive visualisation that is used to evaluate app use in places.

### Place Detection

My Places, described in Section 4.2, integrates with Appwhere to provide place detection with the Kontakt.io LE beacons, displayed in Figure 6.8. The default Kontakt.io settings were used, and My Places was configured to perform LE scans for a duration of 500ms at an interval of 1s. The place estimate is considered to be the beacon with the lowest average calculated distance less than 10m. These thresholds were found experimentally and the system was found to be accurate enough for rapid prototyping. When My Places discovered a new place, a notification was presented in the notification bar. If Appwhere does not receive a place estimate for longer than 5 minutes, the current place estimate is considered to be stale and the device to be in an unknown place, and was labelled as ‘Everywhere Else’. If the device is in an unknown place, and a beacon is detected to be nearby, then a notification alerts the user to add this beacon to My Places.

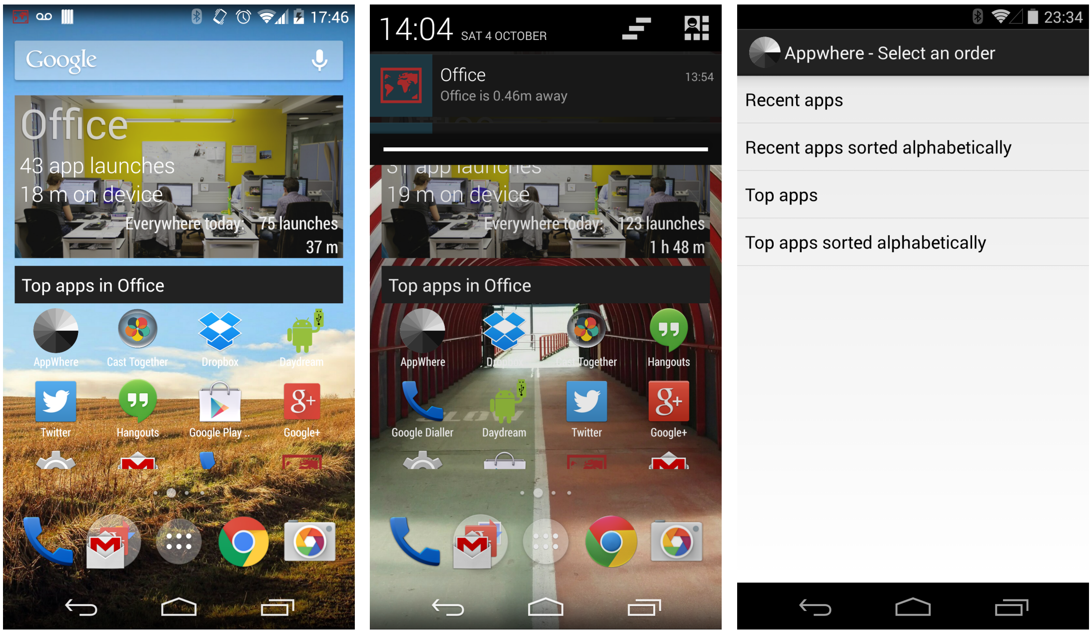



(a) Summary. (b) Most frequently used. (c) Most recently used.

Figure 6.9: Appwhere shares statistics on app use in places with the user. The interface can be swiped left and right to view statistics for each place. Swiping up and down provides an indication of the top apps and most recent apps in each place.

## App Tracking

Since Android 5.0, it is possible to register for app launch events on an Android device. This requires an app to ask for permission to access a user's app launch history and information about the apps in use. As Appwhere was implemented prior to this feature, app tracking is performed by scanning for the list of tasks that are in use and looking for changes to the foreground app. The app tracker scans for changes to the foreground app every 3 seconds, and stores the name and timestamp of the foreground app in a database on the mobile device. There are two main issues with periodically scanning for apps in use: scanning this list will consume power, and so it is desirable to scan as few times as necessary. However, if the scan is not frequent enough, then an app could be opened and closed before it is detected. Therefore, it should be considered what counts as an app launch, and when an app launch is accidental. In the adaptive widget, accidental app launches are not useful when ranking the most used apps on the homescreen. By only scanning the foreground app every 3 seconds, Appwhere considers apps that are used for less than 3 seconds to be insignificant, which was an acceptable assumption for this study. Further investigation into the significance of the duration of an app launch for an adaptive homescreen may be interesting for future work.

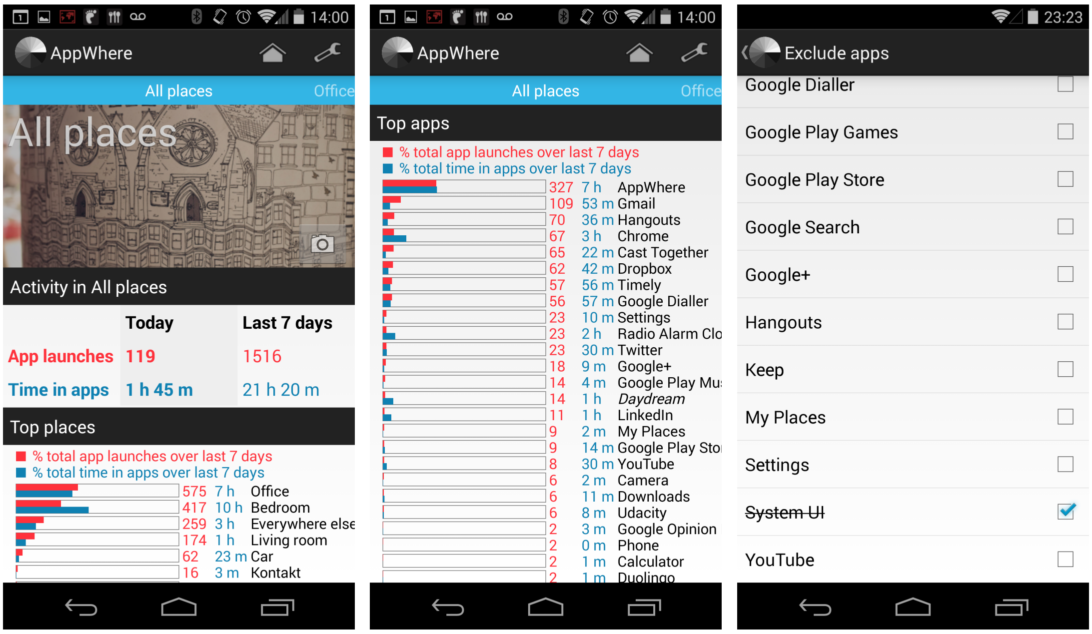



(a) Adaptive Homescreen Widget. (b) My Places notification. (c) Widget order and model selection menu.

Figure 6.10: The Appwhere widget consists of an adaptive widget that fills a page in the homescreen. A summary of statistics are overlayed onto a photo of the place. The order and model of the widget can be selected when the widget is placed onto the homescreen. A notification is displayed to identify the current place, or to notify about beacons nearby that could be added to My Places.

### Adaptive Homescreen Widget

The adaptive homescreen was implemented as a widget that can be placed on the homescreen of an Android 4.0+ device, as displayed in Figure 6.10 (a). This is intended to be a useful tool for finding apps that are likely to be launched in the current place. The text at the top of the widget reflects the label of the place, and at the top of the widget, the photo and summary of statistics update. This helps to recognise which place the device believes itself to be in, and the statistics increase awareness of daily app use. When the user is detected to be in a new place, the database of app launches is queried to retrieve the ranking of apps, and the widget is updated. The widget also updates after each app is launched so that it always reflects the most used apps. Though it is possible to create predictive models of app use, only simple models are considered that are easy to follow: most frequently used and most recently used, which also perform well when the number of app candidates are high [138]. The most frequently used apps can be ranked by the number of times that they have been launched in each place. Most recently used apps can be ranked by the timestamp of when they were last used in a place. A ranking can be chosen for the widget when it is placed on the homescreen, and the top 16 apps can be presented alphabetically or by rank in a scrollable 4x4 grid. 16 apps were chosen to allow increase the accuracy of the simple models.




(a) Time spent in apps in each place. (b) Time spent in each app. (c) Exclude screen.

Figure 6.11: Appwhere shares an overview of app use in places, which could be accessed via the home icon. Apps that should not be tracked could be selected in the Exclude screen, which is accessed with the spanner icon. The photo could be selected by clicking on the camera icon.

### Statistics

Appwhere provides statistics in an app to reflect on where apps are used most, as displayed in 6.9. Statistics are recalculated at the same time as the widget ranking updates. Further statistics were available in the Appwhere app to allow participants to understand where they use apps most, and how their app use differs in each of their places, as displayed in Figure 6.11. If there was any apps that participants were uncomfortable with sharing, they could set this in the Exclude menu, where all apps were ordered alphabetically, as shown in Figure 6.11 (c). This would remove the app from both the widget and the statistics. Some apps were excluded by default as they do not make sense to appear in a homescreen widget. System apps, such as the Android homescreen, System UI and package installer were excluded from the widget by default. These apps are recorded for evaluation purposes.

### Local Data Storage

Appwhere requires app launch data to be stored locally to display usage statistics and to rank the widget. This data can be summarised to conserve space on the mobile device. Appwhere presents app launch history for three time windows: day, week and month (30 days). App launch summaries for the last 30 days are stored in a SQLite database, which can be queried to rank apps by most recently used or by most frequently used. App launch summaries for each day of the week are also stored locally to be displayed as daily and weekly statistics. At the end of each day, the daily summary is cleared, along with data that is older than 30 days. Individual app launches, notifications and widget launches are interesting for analysis. These are stored on Google Cloud Datastore.

### Data Storage with Google Cloud Platform

There are limitations to storing data locally on the smartphone, including accessibility of the data during the experiment, reliability of the data in the event of a smartphone failure, and also storage constraints. This approach will not scale over time and the number of participants. Therefore, it is desirable to store data on a remote server. Web-based service providers such as Google Cloud Platform and Amazon Web Services make it simple to develop on top of platforms that scale.

Google Cloud Platform provides a set of modular services to build an application. Examples include AppEngine, Cloud Endpoints, Datastore and Compute Engine. AppEngine is a Platform-as-a-Service that can host a web application. Cloud Endpoints can generate RESTful services to integrate with Android. Datastore is a NoSQL database for storing non-relational data and performing SQL-like queries (GQL). Compute Engine hosts virtual machines on Google's infrastructure that can be set up run large-scale analysis on big data sets. Appwhere is built on Google Cloud Platform to access these services. Though Google Cloud Platform is pay-per-use, only the free quota was used in this study.

Individual app launches, notifications and widget launches are stored on Cloud Datastore, which requires connection to the internet. As data is not required in real-time, data is only send when the device is connected via Wi-Fi, and storage is rate-limited by batching the data and only sending entries every 1 second. When data has been successfully stored in Datastore, it is deleted from the device.

Cloud Endpoints was used to generate a client library for the Appwhere Android app. This made it simple to send data to Datastore. Endpoints was also used to pull monthly data from Datastore to be summarised in the statistics, in the event that Appwhere was reinstalled. Users were identified simply by the unique id of their mobile device, though it would be possible to create a user account to cope with the possibility of changing smartphone during the study.


Figure 6.12: A visualisation of the app usage data over one week for one participant.

### Interactive Visualisation

An offline interactive visualisation was created to explore and discover insights in the app use data. This is displayed in Figure 6.12. The visualisation was implemented with D3.js,[1] which allows the data to be filtered in real-time. This provides an overview of the app launch behaviour, including the time of day, places, and days when app use was most frequent. App launch data from the last month can be downloaded from the remote server, or alternatively, data could be imported from a json file.

The interactive visualisation is built on AppEngine, which was hosted locally. This uses Cloud Endpoints to retrieve app launch data from Datastore to be explored. As Datastore supports GQL queries, data can be easily retrieved by device id and time period. GQL queries were used to filter the datastore entries, and the amount of data that was loaded into the visualisation was limited to entries of a single device over a period of one month. This data was exported to a json file so that is can be read from the local file system. A backup of the datastore entries was also created to export the data. This was used to iterate over the datastore entries locally to analyse the accuracy of the widget.

While the study was in progress, app data from the past 30 days was viewed in the visualisation. It would be simple to introduce user accounts and host this application online to allow users to explore their personal app launch data. However, scaling this application to a larger number of users would introduce potential ethical concerns, including how to gain consent from end users [108]. Additionally, storing a higher quantity of app launch data could increase beyond the free quota provided by the Google Cloud Platform. Therefore, launching the interactive visualisation in the wild was not explored in this project.

## 6.2.2 User Study: Evaluating Appwhere in Places

A user study was run to evaluate the updated design of Appwhere, with Bluetooth LE beacons to provide more robust place detection. The use of the adaptive homescreen menu, and insights from app use data in personal places, was evaluated for 5 participants over a period of 5 months.

### Research Methods

Appwhere and My Places were installed on the Android smartphones of 5 participants, and their launch history was recorded over a period of 5 months. LE beacons were provided to track app use in personal places. As LE beacons were only supported by newer hardware,

[1] `http://d3js.org/`

participants with older smartphone models were provided with a Nexus 5 smartphone for the purposes of this study. Participants were encouraged to use this smartphone as normal.

Participants were invited to a Google+ Community to become a beta tester of the experiment app, and an information sheet provided instructions to download the Appwhere and My Places from the Google Play Store and to set up the Kontakt.io beacons with the My Places app. At the start of the experiment, participants were requested to place the adaptive widget on the homescreen, and were given the option of how the widget should be ranked and ordered.

At the end of the study, participants were asked to fill out a questionnaire about their experience. This questionnaire is available as Appendix E. Participants were helped with uninstalling the experiment applications, and the beacons were collected. As app use data was stored on a server when the device was on a Wi-Fi connection, this could be loaded into the visualisation tool during the study. Therefore, no further data collection was needed at the end of the study.

#### Participants

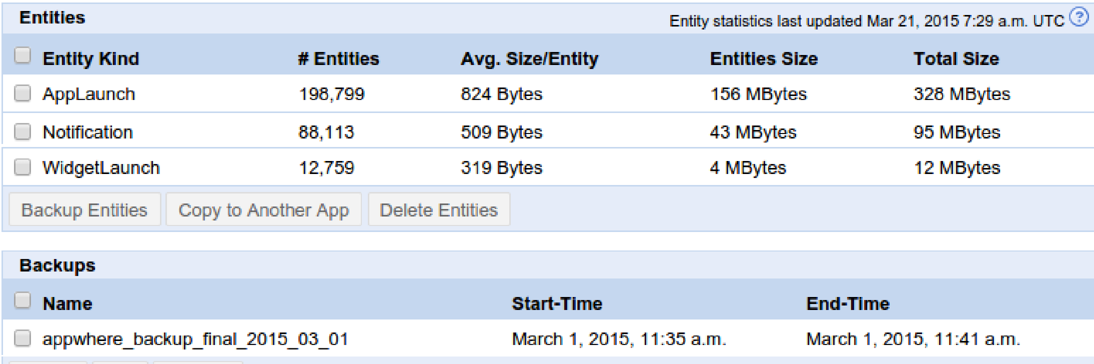
5 participants took part in the experiment, aged between 29 - 62, and one was female. Three participants took part in the initial study of Appwhere from Section 5.4. Participants were asked to place an LE beacon in five personal places where they used their smartphone most. All participants chose to put beacons in their Living room, Bedroom, Kitchen and Car. Participants who placed the beacon in a car were reminded not to use the smartphone while driving. Some participants with an Office also chose this as a place. A Summerhouse, Bookshelf and Dining room were also chosen as places. All participants were familiar with at least one other participant, and shared at least one place with another participant during the experiment.

### 6.2.3 Results

The analysis of app launch behaviour, notifications and the adaptive homescreen widget is presented, in addition to feedback related to the app use statistics and the rapid-prototyping approach to place detection. The subjective opinions of participants received from questionnaires are discussed in relation to each section.

#### Place Detection

Participants were asked to comment on the set up of the place detection with Bluetooth LE beacons and the My Places tool. The Bluetooth beacons were found to be “small and



Entities — Entity statistics last updated Mar 21, 2015 7:29 a.m. UTC

| Entity Kind | # Entities | Avg. Size/Entity | Entities Size | Total Size |
|---|---|---|---|---|
| AppLaunch | 198,799 | 824 Bytes | 156 MBytes | 328 MBytes |
| Notification | 88,113 | 509 Bytes | 43 MBytes | 95 MBytes |
| WidgetLaunch | 12,759 | 319 Bytes | 4 MBytes | 12 MBytes |

Backup Entities | Copy to Another App | Delete Entities

Backups

| Name | Start-Time | End-Time |
|---|---|---|
| appwhere_backup_final_2015_03_01 | March 1, 2015, 11:35 a.m. | March 1, 2015, 11:41 a.m. |

(a) Google AppEngine Datastore Admin of Appwhere.

| Event | Total (All Places) | Total (In Places) | % In Places |
|---|---|---|---|
| App Launch (non-system) | 25,492 | 11,747 | 0.46 |
| Screen off | 18,198 | 9,749 | 0.54 |
| Notification dismissed | 25,443 | 14,043 | 0.55 |
| Widget Launch | 2,362 | 1,441 | 0.61 |

(b) Summary of events used in the analysis.

Figure 6.13: Summary of data collected.

discreet" [P3] and were "not a distraction" [P4]. Though participants managed to set up the beacons with My Places, P1 "felt it was a little complicated while setting up initially, though when it was set up, I found it very helpful".

The place detection was found to be unreliable at times. P1 thought that "most of the time it was very accurate, but would on occasion jump to an adjacent space if I strayed too far from the beacon in the place I was in". Though the My Places notification was not noticed by participants, it provided a way to check the place detection: "Mostly I would not notice the notification. When I did, I felt reassured that the beacons were working as intended" [P1]. The photo also increased awareness of which place the device was in. Participants chose photos from their camera, and P1 "used photos and images that reminded me of, or were taken in the place I was assigning them to". The photo was found to be "interesting as it told you where you were, but you already knew" [P3] and "very useful, as I could see at a glance if my phone was being effected by the space I was in" [P1].

When asked if participants would have liked to track additional places, most participants were happy. P3 felt that "No I think that was enough", and P1 shared "I feel I was pretty well covered for places I spend most time; Kitchen, living room, bedroom, car, office. All other places fell quite comfortably into the Everywhere else category". P4 would have liked "a travel one in a toddler group or play areas I visit on a regular basis". As Bluetooth beacons become more pervasive in public places, it will be possible to assign public places to My Places. However, it would be desirable to use other device sensors, such as Wi-Fi and

GPS, to detect a larger number of places.

### Self-Reflection

P1 checked the statistics “no time in particular, simply when I was procrastinating and had remembered that it was there... I found them interesting, though did not regularly check them. It was sometimes a surprise to see how much time I had spent on my device, though I don’t think knowing that information directly effected how much time I subsequently spent using my phone”. P1 launched Appwhere 119 times, in all his places, including Elsewhere (42), Living room (27) and Bedroom (21), and spent total duration of 72.5 minutes in Appwhere, with an average visit lasting 36.5s.

P1 “noticed that there was a definite relation between apps I used and the places I used them. I had originally thought that I used the same few apps across all places, but many of them turned out to be specific to one place when looking at them as an overview’. P1 would also have liked to see “historic app usage, or an ‘all time’ app use summary instead of just the 7 days”.

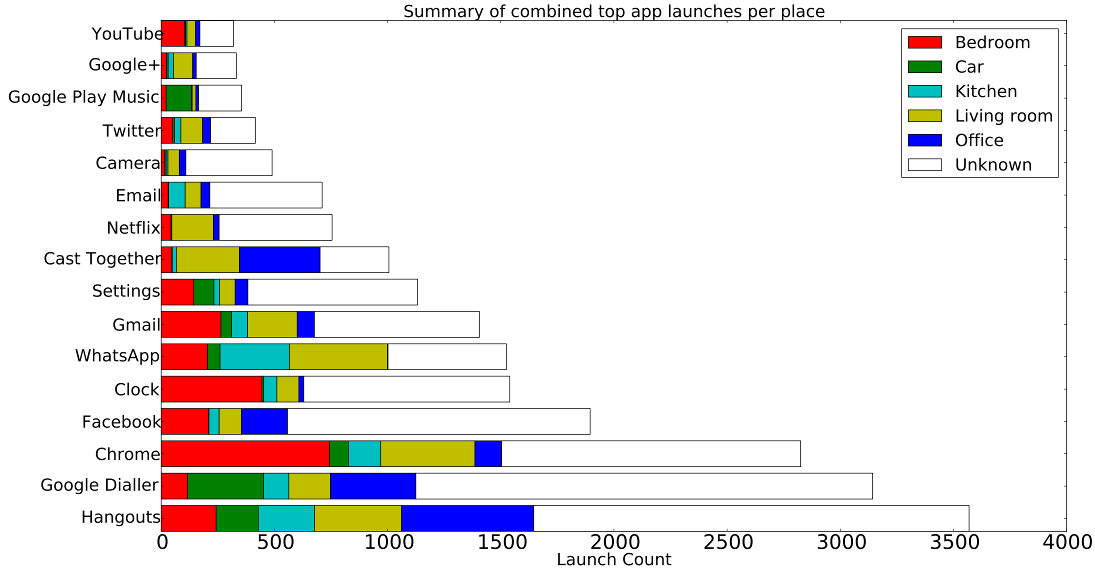
P5 checked Appwhere 21 times, mainly between 4 - 6pm, and spent a total duration of 19 minutes. P3 launched Appwhere 10 times, and found the statistics “made you aware of what apps you used most frequently” and “found it to be very enlightening [to see the number of] apps used and where I was using them”.

Overall participants “found the whole experiment interesting and helped me understand my smartphone and app usage” [P3] and “it was interesting and I would like to continue using it” [P1].

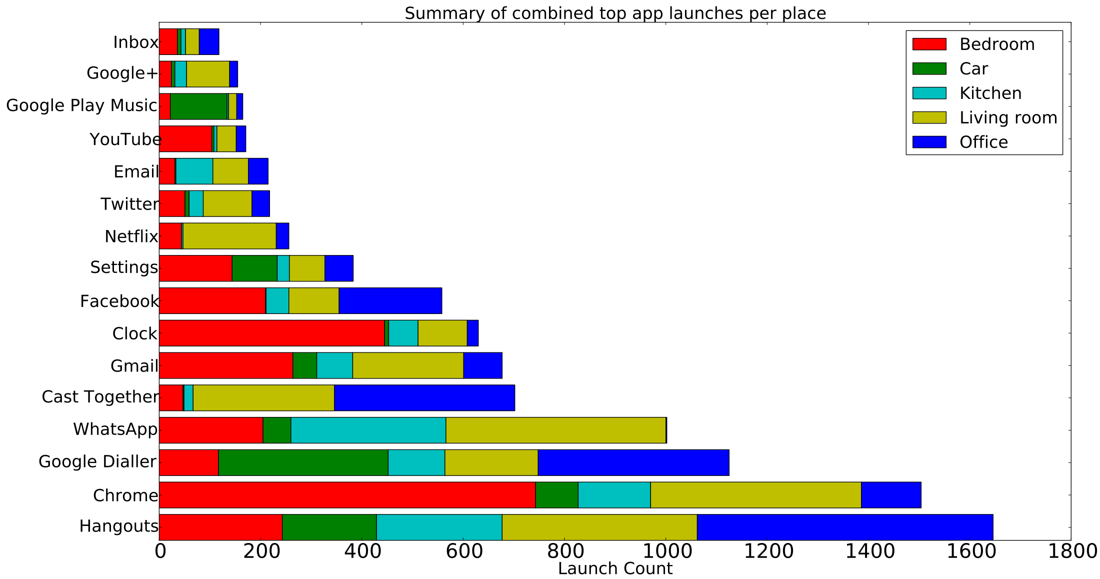
### Information Needs

Almost 300,000 entities were gathered with Appwhere, as visualised in the Datastore Admin screenshot in Figure 6.13 (a). 25,492 app launches and 18,198 screen off events were recorded for 5 participants over a period of 5 months, between 1 Oct 2014 - 25 March 2015. 11,747 app launches and 9,749 screen off events were detected in places, accounting for 46.2% of app launches and 53.6% of smartphone sessions. Appwhere also tracked 25,443 notifications, and 14,043 were detected in places (55.2%).

Figure 6.14 (a) and (b) display the top 16 apps for all participants. It can be seen that the Camera app is more popular everywhere else than in personal places. In places, Netflix is more popular in the Living room, and Google Play Music is most popular in the Car. Figure 6.14 (c) provides a summary of app launches per participant overall and in personal places. P1 recorded the most app launch events (37%) followed by P3 (24%).




(a) All places.




(b) In places.

| PID | Period (days) | Total (All) | Mean (day) | % of Events | Total (Places) | Mean (day) | % in Places |
|---|---|---|---|---|---|---|---|
| 1 | 177 | 9,429 | 53 | 0.37 | 5,182 | 29 | 0.44 |
| 2 | 79 | 4,860 | 62 | 0.19 | 1,295 | 16 | 0.11 |
| 3 | 72 | 6,065 | 84 | 0.24 | 3,179 | 44 | 0.27 |
| 4 | 123 | 1,533 | 12 | 0.06 | 530 | 4 | 0.05 |
| 5 | 81 | 3,605 | 45 | 0.14 | 1,561 | 19 | 0.13 |

(c) Total share of app launch events per participant overall and in places. Totals do not include system apps, such as the Launcher.

Figure 6.14: Summary of all participant app launches.

### Adaptive Homescreen

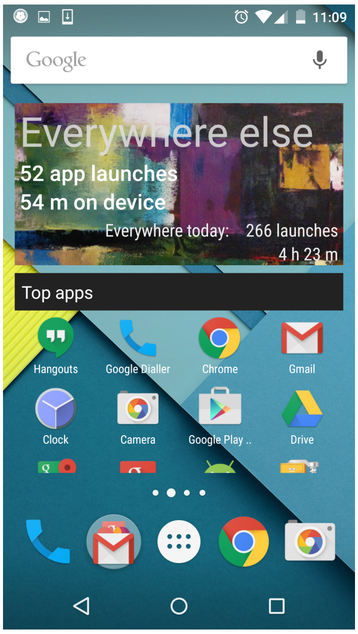
25,492 app launches by all participants were non-system apps, and 2,362 were from the adaptive widget (9.3%). 1,441 apps were launched from the widget, accounting for 61.0% of all widget launches.

The homescreen widget was used by all participants. P3 "found it useful in making the most used apps easier to access" and P1 shared that he "Loved it. Before, my home screen had all of the apps I was ever likely to use, and for the most part, in no particular order. Appwhere allowed me to clear the homescreen, and once set up, I was often unaware of using it". P1 also shared that "I found it useful almost every time I interacted with my phone. For example, when entering my car - I would be presented by play music app as a first suggestion. This is also the case for most other places, and I do feel it made my interactions with my phone more efficient". Figure 6.15 (e) displays the top apps for P1 in the Car. Google Play Music app is shown at position 2. The adaptive widget allowed P1 to have "far less applications on my homescreen. In the past I used around four screens which I navigated through to find the app I wanted. With Appwhere, the majority of app launches came from a single main homescreen".

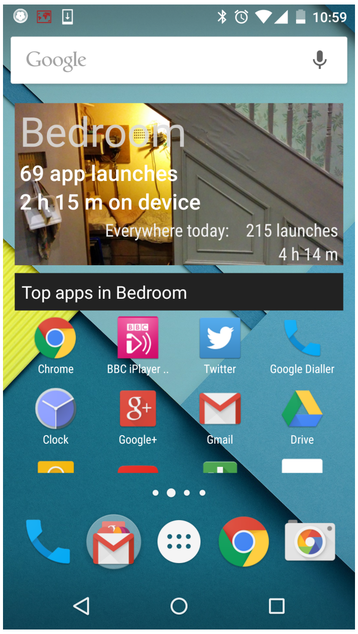
**Accuracy**

To consider the accuracy of the widget, the app use data of a single participant is examined. P1 launched 1,503 app launches from the widget, and 1,098 were in places (73.1%). Table 6.16 summarises the top 4 apps in each place for P1, and Figure 6.15 displays his homescreen widget in each place. P1 used a total of 57 apps: Living room (45), Unknown (44), Bedroom (41), Office (30), Car (24), Kitchen (23). The majority of his apps were launched Elsewhere (45.9%), followed by Bedroom (17.8%) and Living room (15.2%).

To demonstrate the value in adapting to places, the accuracy of adapting to places can be compared to a static split menu with 4 apps. Compared to an adaptive menu, a static menu requires prior knowledge of the frequency of app launches. With perfect knowledge of how many times P1 would launch each app, it would be possible to rank the overall top 4 apps: Chrome, Clock, Dialler and Hangouts. This menu would be 54% accurate for P1. The accuracy of this static menu in places can be calculated by taking the number of times that apps in the static menu were launched in each place, and compare this to the total number of apps launched in each place: Elsewhere (60%), Bedroom (51%) and Living room (32%). This static split menu can also be compared to a split menu that adapts to places. For each place, the adaptive static menu would contain the top 4 apps used in each place. For P1, this menu would be: Elsewhere (60%), Bedroom (62%) and Living room (59%). This simple example demonstrates the value in adapting to places. Compared to a static menu, Appwhere is a dynamic menu that presents the top 16 apps in a scrollable split menu. As it is not

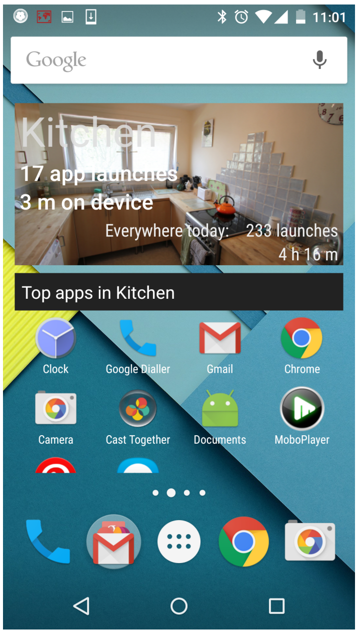



(a) Elsewhere.

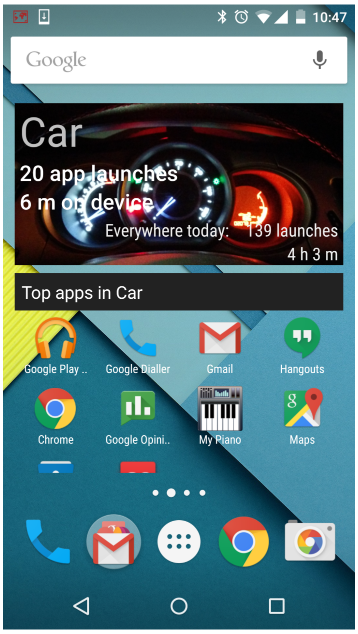



(b) Bedroom.

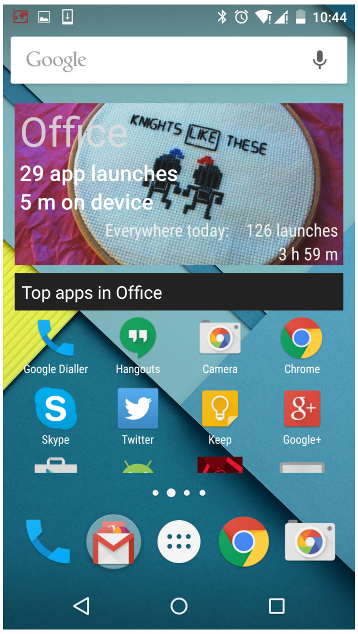

(c) Living room.




(d) Kitchen.

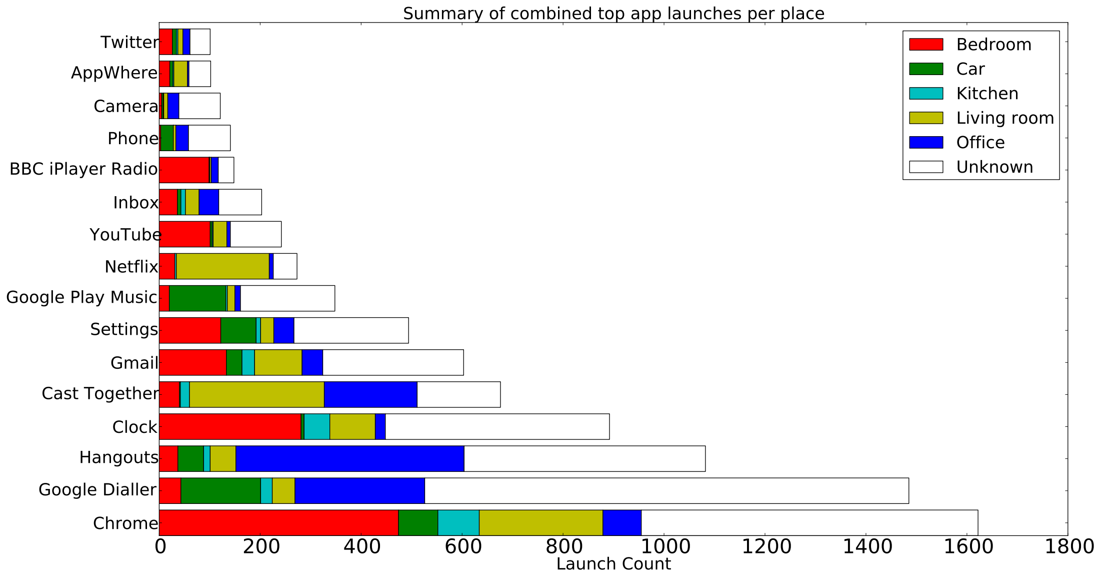



(e) Car.

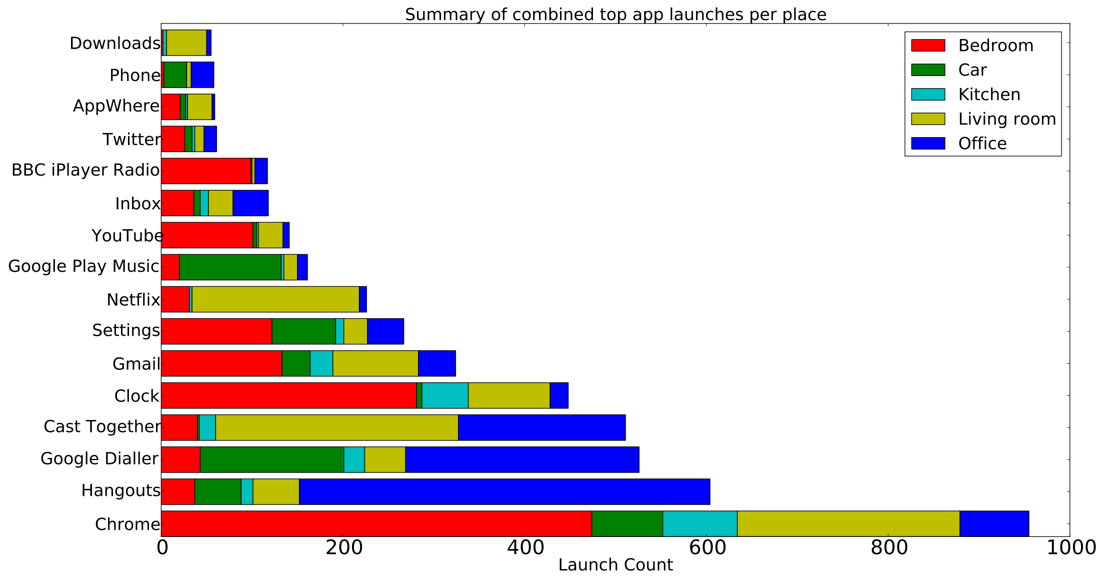



(f) Office.

Figure 6.15: Screenshots of the adaptive widget for P1 in each of his places taken after the experiment. Note: the Cast Together application displayed in the Living room and Kitchen is presented in Chapter 7.




(a) All Places.




(b) In Places.

| Place | Total | Accuracy (In Places) | Accuracy (All Places) | Top 4 Apps (alph) |
|---|---|---|---|---|
| Office | 1,281 | 0.76 | 0.63 | Cast Together, Chrome, Dialler, Hangouts |
| Kitchen | 277 | 0.69 | 0.65 | Chrome, Clock, Dialler, Gmail |
| Car | 683 | 0.67 | 0.47 | Chrome, Dialler, Music, Settings |
| Bedroom | 1,682 | 0.62 | 0.51 | Chrome, Clock, Gmail, Settings |
| Elsewhere | 4,328 | 0.60 | 0.60 | Chrome, Clock, Dialler, Hangouts |
| Living room | 1,431 | 0.59 | 0.32 | Cast Together, Chrome, Gmail, Netflix |
| *All Places* | *9,429* | *0.54* | *0.54* | *Chrome, Clock, Dialler, Hangouts* |

(c) Comparison of the accuracy of the top 4 static apps In places vs All places.

Figure 6.16: Static example. Top non-system apps used in places by P1.

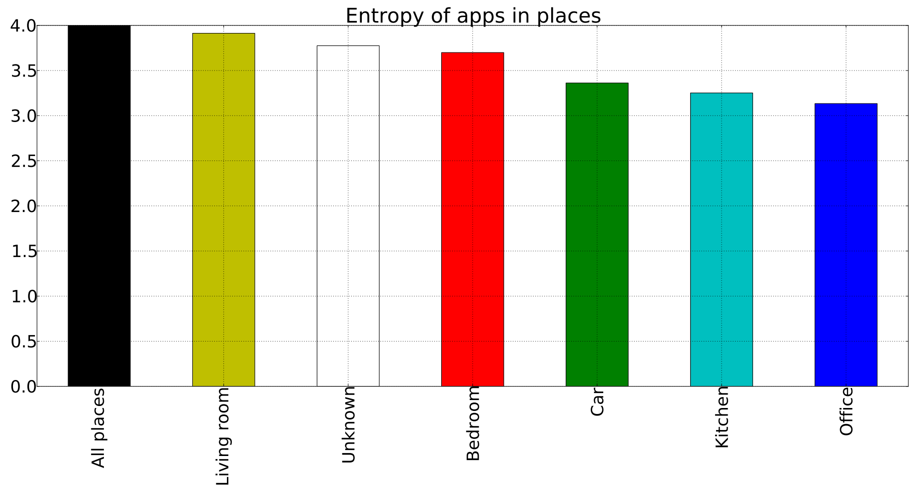



Figure 6.17: Entropy in app launches per place for P1: All places (black), Unknown (white), Living room (yellow), Bedroom (red), Car (green), Kitchen (cyan) and Office (blue).

possible to have perfect knowledge of how many times each app will be launched in places, an adaptive menu accumulates this knowledge over time.

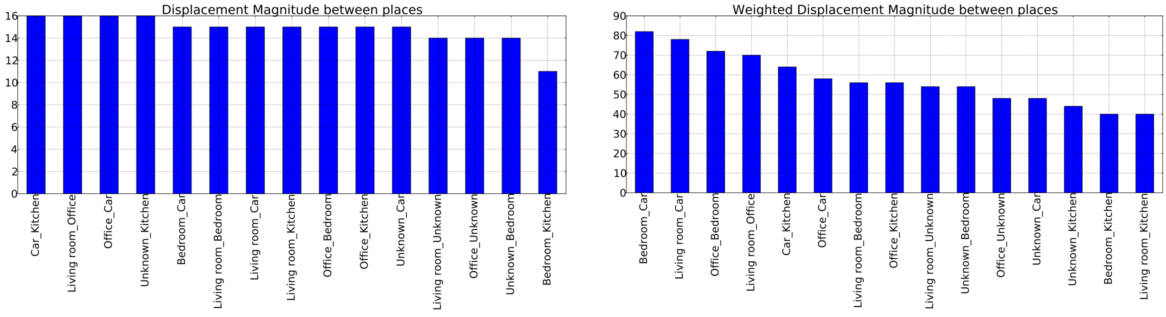
### Entropy

Entropy in information theory is a measure for how mixed a set is. The aim of the adaptive menu is to reduce the amount of entropy in the selection. The entropy of apps used in places is measured to consider how differently apps are used between places. Figure 6.17 displays the entropy in each place for P1. The entropy of the app launches in places decreases compared to the overall entropy in all places. Entropy is lowest in the Office, and highest in the Living room. Therefore, the widget is expected to be more accurate in the Office than in the Living room.

### Adaptations

P1 shared that "[the adaptations were] my favourite part of the widget. It really did tailor my home screen to the different situations I find myself in, but for the most part, did so without me thinking about it". Adapting to places "was most useful for specific apps that I used regularly in certain places. E.g. Timer in the kitchen and BBC iPlayer radio in the bedroom". Figure 6.15 displays the top apps for P1 in the Kitchen (d) and Bedroom (b). The Clock app is at position 2 in both places, and the BBC iPlayer Radio is at position 6 in the Bedroom. P1 also "realised that I have a habit of downloading apps with Google Play Store as I wait in my partner's office. Games that I did not recognise were also indicated as Top Apps, that I must have downloaded but did not have time to play. This was a very useful suggestion to pass the time".

P1 also found times when the adaptive changes were frustrating: "The majority of the time I didn't notice the order changing. Usually, if I unlocked my phone and noticed the app I




(a) Displacement Magnitude.

(b) Weighted Displacement Magnitude.

Figure 6.18: UI displacements for P1 when moving between places.

was intending to use anywhere on the Appwhere widget, I would select it without a thought to it's position. The only time I noticed the apps updating was if they updated while I was looking at them. Occasionally this would happen right at the moment of selecting an app, causing me to launch an app that I didn't intend to. This was really quite frustrating, and the only real negative experience I had with the widget".

**UI Displacements**

UI displacements can be measured to understand the magnitude of app movements in the widget as P1 moves between his places. Figure 6.18 displays the displacement magnitude (DM) and weighted displacement magnitude (WDM) for P1 moving between each of his places. The DM shows that more than 10 apps change position when transitioning between all places. The WDM shows that changes occur towards the start of the list more when transitioning between either the Car or Office to the Bedroom or Living room. Fewer UI displacements occur when moving between the Kitchen and either the Bedroom, Living room or an Unknown place, or when moving from the Car or Office to an Unknown place. Perhaps surprisingly, the widget will update more when transitioning between the Living room and Bedroom, than transitioning from either to the Kitchen. The top apps used in these places can be seen in Figure 6.15: The Clock is used in both the Bedroom and Kitchen, and apps such as Netflix are shared between the Kitchen and Living room.

### 6.2.4 Discussion

Appwhere, a prototype for tracking app use in personal places, was presented with an updated design to improve place detection with Bluetooth LE beacons, and increase awareness of app use in places by presenting app use statistics on the homescreen. A user study was performed with 5 participants over a period of 5 months. Despite the small number of participants in this user study, the rapid-prototyping approach demonstrates an approach to using logical labels to adapt the homescreen to contextual data that is easy to understand. Updates to the My Places tool and the use of Bluetooth LE beacons made it possible to perform a longitudinal user study and gain quantitative insights into app use in places. Feedback from participants and quantitative data of app launches from the adaptive homescreen menu show that it is possible for an adaptive menu on the homescreen to blend in with menu navigation habits. It was also found that participants better understood their app use behaviour in places by reflecting on the app launch statistics. However, the small population of participants were not motivated to act on this awareness, and further work is required to evaluate the impact of self-reflection on app launch habits more fully.

Though Appwhere focuses on personal places, other contextual factors could become the focus of an adaptive homescreen, including social contacts who are nearby. Furthermore, social awareness could be encouraged by sharing app use statistics with friends, to compare mobile habits and motivate change. It would also be interesting to consider changing the focus of the adaptive homescreen, not only to different places, but also to social situations, or situated activities that are important to the user. For example, a menu that has adapted to apps used in an office could change to present more fine-grained tools related to an activity at a desk, or might enter a collaborative mode when the user starts to engage with a colleague. The potential to incorporate situated interaction and social collaboration in the adaptive menu presents an interesting direction for future work.

## 6.3 Data-Driven Choice in App Navigation

On Android smartphones, several choices exist for navigating between apps on a smartphone: The homescreen offers a subset of installed apps to launch, which can be manually configured by the user (adaptable) or predicted automatically (adaptive). The homescreen menu consists of multiple panels that can be swiped horizontally, and a dock containing a small number of apps is accessible from all homescreen panels. Apps can also be organised on the homescreen in folders. The full list of installed apps can be accessed in the app drawer, which can be swiped horizontally or vertically. A search interface can also be used to navigate apps, including text search, voice search and gesture search. The variety of interfaces for app navigation provide a choice to the user, and it is not always clear which choice provides the most value. It is useful to understand when a change in user interface is appropriate, particularly in the many contexts in which the smartphone can be used.

App navigation on Android mobile devices has been studied by [66], and it was found that the majority of apps are started from within other apps, followed by the use of the homescreen. Comparing the average navigation time of the different launch types, their study showed that the dock was the fastest (2.24s), followed by homescreen panels (2.65s), and that folders (4.66s), vertical (5.55s) and horizontal app drawer (7.21s) were the slowest. [66] also found that users have a good self-assessment about navigation speed, and that they will choose the menu that is most efficient to complete a navigation task. Economic models have been applied to interaction to predict and explain user behaviour, as demonstrated with a search interface [5].

With a record of app use in user contexts, it is possible to evaluate data-driven models of app navigation. Data-driven models of app navigation are explored as a way to evaluate the choice in app navigation. Models of app navigation are specified that estimate the time to select apps from: the app drawer, the homescreen and a search interface. Each model is evaluated with increasing numbers of installed apps and increasing model accuracy to demonstrate the quantitative approach. The model is then extended to consider contextual factors that impact choice in app navigation.

## 6.3.1 Costs in App Navigation

In app navigation, there are several relevant measures that are associated with all navigation menus, including the number of installed apps to filter and how quickly one can arrive at the menu interface. Several measures will only apply to certain forms of navigation. For example, for a text-based search, typing letters will incur a cost, and the number of letters entered will determine the number of suggestions to be inspected. In comparison, the app drawer does not require typing. The homescreen menu will have an accuracy that will determine how often an app will be present, and how often the user will be required to use a secondary navigation mode.

### Parameters

The parameters of app navigation are summarised as follows:

- All menus:
  - the number of installed apps, $N_A$.
  - the number of interactions to access the menu, $N_I$.
  - the number of candidates that are displayed on a single screen, $N_C$.
  - the number of screens to display (pages to swipe or scroll), $N_S$
  - the time to perform a single interaction (swipe, key press), $t_I$.
  - the time to update the screen, $t_u$.
  - the time to inspect a candidate, $t_{in}$.

  - the familiarity of the user with the app name and icon (frequency and recency of access), $P(r)$.
- Text-based search:
  - the number of apps reduced per letter, $R$.
  - the number of letters entered, $N_L$.
- Homescreen:
  - the accuracy of the homescreen menu, $Acc$.

### Cost Function

There are many cost factors that can impact app navigation, including the time, effort and skill required to find an app with a particular menu and in a given context. For all navigation menus $(C_M)$, there will be a cost to open the menu $(C_O)$, a cost to filter the app candidates $(C_F)$, and a cost to find the target app in these candidates $(C_T)$. A basic cost function might look as follows:

$$C_M = C_O + C_F + C_T \tag{6.5}$$

The cost to open the navigation menu will depend on the number of interactions to access the menu and the cost of each interaction:

$$C_O = N_{I_M}.C_I \tag{6.6}$$

When the app drawer and homescreen menus are open, the number of suggestions $(N_C)$ will depend on the number of apps that the interface can display. For the search menu, $N_C$ will depend on the number of letters entered and the reduction of apps per letter:

$$N_{C_{search}} = \frac{N_A}{R^{N_L}} \tag{6.7}$$

The cost of filtering $(C_F)$ the number of suggestions will depend on the form of navigation. Filtering the app drawer requires swiping between its pages. As the number of pages in the app drawer $(N_P)$ depends on the number of installed apps, the cost is expected to increase with the number of installed apps. On average, the cost of filtering the app drawer is expected to be as follows:

$$C_{F_{Drawer}} = \frac{1}{2}\left(N_P.C_I\right) \tag{6.8}$$

Filtering a search interface requires typing letters, and depends on how much the search function reduces the number of suggestions per letter. On average, the cost of filtering is expected to be as follows:

$$C_{F_{Search}} = \frac{1}{2}\left(N_L.C_I\right) \tag{6.9}$$

A homescreen filters app suggestions without interaction, and depends on the accuracy of the predictive model. As it is possible that the app is not predicted by the homescreen, and is not available in the menu, then there will be a cost of using another navigation menu $(C_M)$. On average, the cost of filtering is expected to be as follows:

$$C_{F_{Homescreen}} = (1 - Acc).C_M \tag{6.10}$$

Finally, the cost to find a target app $(C_T)$ will depend on the number of suggestions to inspect after a filtering step, and the time to inspect a single app icon. This inspection step might be performed multiple times depending on the familiarity of the user with the app icon $(P(r))$, and the maximum number of screens to inspect $(N_S)$. On average, the cost of finding an app can be estimated as follows:

$$C_T = \frac{1}{2}.N_C.C_{in}.(1 - P(r)).N_S \tag{6.11}$$

The costs to access the menu, filter the suggestions and find the target app can be added to calculate the total cost of the navigation menu for a given set of parameters. This basic model is used as a starting point to gain insight into the choice associated with app navigation.

### 6.3.2 Cost Analysis: Impact of Accuracy and Installed Apps

To illustrate the cost function, example parameters are provided to compare the optimal navigation menu as the accuracy of the homescreen and number of installed apps increase. Three navigation menus are compared: app drawer, text-based search, and homescreen. The parameters of their cost functions are based on those found in the literature.

Two dependent parameters are compared: the number of installed apps [50, 100, 150], and the accuracy of the homescreen [0.25, 0.5, 0.75]. Figure 6.19 displays the values that are associated with the independent parameters. It is assumed that the app drawer will be used if the app is not contained in the homescreen $(C_{Drawer})$. Based on these parameters, it is considered when the homescreen might be an optimal navigation mode.

Figure 6.20 displays the results of the cost function using parameters for the app drawer, text-based search and the homescreen as the number of installed apps and homescreen accuracy

| $N_{I_{menu}}$ | $t_I(s)$ | $t_u(s)$ | $t_{in}(s)$ | $P(r)$ |
|---|---|---|---|---|
| 1 | 0.4 | 0.2 | 0.4 | 0.8 |

(a) All menus

| $N_C$ |
|---|
| 25 |

(b) App drawer

| $N_L$ | $R_L$ |
|---|---|
| 5 | 20 |

(c) Search

| $N_C$ | $C_M$ |
|---|---|
| 8 | $C_{Drawer}$ |

(d) Homescreen

Figure 6.19: Model parameters

increase.

The cost of the app drawer is between 5s and 8s depending on the number of installed apps. This fits with the expectations from [66], where the horizontal app drawer was found to take 7.21s on average. A homescreen that is 75% accurate has an estimated cost of between 2s and 3s, which also fits with the findings of [66] (2.65s). However, as the accuracy of the homescreen decreases, the cost of navigating apps approaches the cost of the app drawer, which is chosen to be the secondary menu.

The accuracy of the homescreen has an impact on the best navigation mode. If the homescreen contains at least 3 in 4 apps (75 % accurate), then it will consume the least time on average. In comparision, a homescreen that contains 1 in 2 apps (50% accurate) will depend more on the app drawer, and will be as costly as the search menu. When the homescreen has low accuracy (25%), it will perform as badly as the app drawer on average. Therefore, the homescreen can provide a faster method of navigating apps when its predictive model is accurate or when there are few installed apps.

The cost of the app drawer increases with the number of installed apps. In comparison, the number of installed apps has little effect on the search menu. Therefore, search might be more cost effective as a secondary menu than the app drawer.

### 6.3.3 Impact of Context in App Navigation

The previous section demonstrates an approach to analysing the cost of app navigation from a set of parameters, based on existing work in economic models and cost functions. It is interesting to extend this approach, and consider how a simple model of app navigation could improve the design of app navigation menus in the context of places. The choice of which app navigation menu to use will depend on a larger number of parameters than those considered in the simple example given. Additionally, the values of each parameter will depend on the personal experience of the smartphone user, and will change according to their context in the user environment. The physical effort of a navigation menu are additional

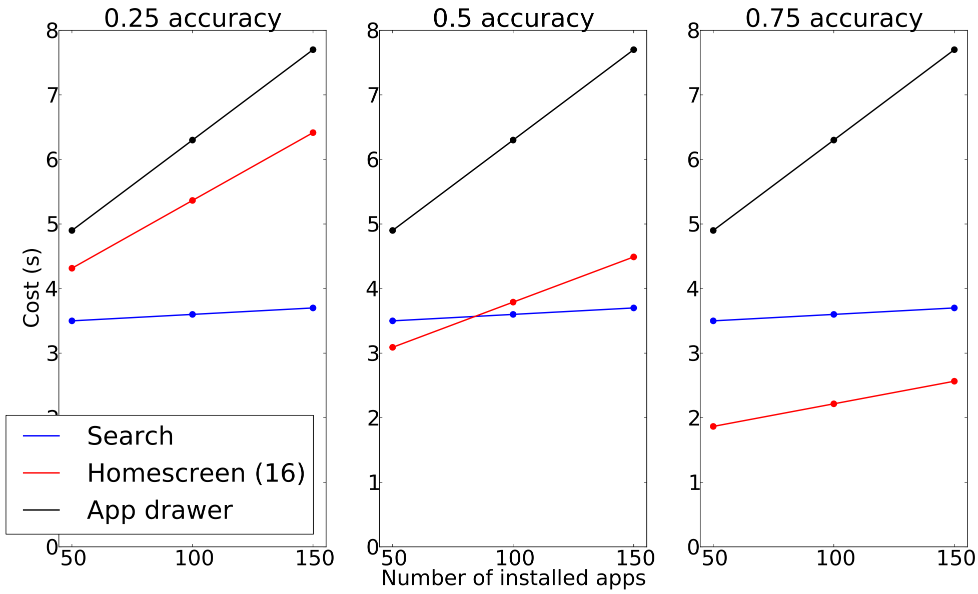



Figure 6.20: The average cost of homescreen with increasing accuracy and number of installed apps, compared to using the app drawer or search.

parameters that could be modelled, in addition to the accessibility and social acceptability of each navigation menu.

Given the user context, features of the data-driven model could become more important than others: If a user is encumbered by holding a box, then a hands-free interface could become less costly than a menu that requires touch input. Social acceptability may have a bigger stake in menu interface when the user is in a public setting, and a subtle navigation mode could be more socially acceptable in a quiet co-located environment, compared to a voice-activated menu that would cause a disturbance to other people.

The parameters of the data-driven model could be weighted according to their importance in a given context. Furthermore, each step of the navigation task - access, filter and find - could be weighted according to their importance, as follows, where $\alpha_i$ implies a weighting:

$$C_M = \alpha_1.C_O + \alpha_2.C_F + \alpha_3.C_T \tag{6.12}$$

For example, if the app should be found quickly, then the cost function could be minimised to find the shortest selection time; if it is important that the interaction is discreet, then a longer selection time might be acceptable. By collecting data about menu navigation modes in places, it would be possible to compare the cost of menu navigation beyond selection time, and design choices that support a wider set of contexts.

### 6.3.4 Discussion

A data-driven evaluation of choice in app navigation was explored by considering the costs associated with three interfaces for app navigation: app drawer, search and homescreen. The parameters associated with these navigation menus were identified, and a cost function was presented that sums the costs to access the menu, filter the suggestions and find the target app. Grounding assumptions on prior data from studies of navigation menus, the approach was demonstrated by comparing the impact of increasing the number of installed apps and the accuracy of the homescreen on the cost of each menu, and considered the circumstances when the homescreen might be used. With the parameters that were defined, it was shown that the app drawer can be a costly form of app navigation, especially as the number of installed apps increase, which fits with current knowledge. In comparison, a search menu is less dependent on the number of installed apps. The homescreen depends on a secondary menu that will be utilised if the app is not predicted by its model, and this dependency weakens as the homescreen becomes more accurate. This demonstration shows that the homescreen can provide a faster method of navigating apps when its predictive model is accurate or when there are few installed apps. Limitations of this approach include the assumptions made in modelling the navigation menus, and the small number of parameters defined. Future work will include extending these models and validating the assumptions made with a user study.

## 6.4 Summary

This chapter focused on a place-aware adaptive homescreen, and considered how it might improve app navigation and increase awareness of app use in personal places (RQ-2 of Section 3.1).

A challenge of adaptive menus is that adaptations can be difficult for users to follow. An evaluation was performed to evaluate the stability of an adaptive menu, by controlling the ordering of the adaptive homescreen menu as the adaptive model becomes less easy to understand. The time to complete a selection task with the adaptive menu was found to be weakly correlated to the displacement of items in the menu. Selection time and subjective rating improved significantly when both the model was easy to understand and an alphabetical order was used, which were conditions that increased menu stability. However, for a model that was less easy to understand, ordering by the rank of the model provided a more usable search strategy, as the most likely items were positioned near the start of the menu.

Insights gained from studies with the Appwhere prototype were used to iterate on the design of the adaptive homescreen menu and interactive visualisation. A user study was performed with the refined prototype, which focused on app use in personal places and self-reflection on app launch behaviour. Bluetooth LE beacons were used to detect places, which improved the usability of the My Places tool. Participants of the study reported that the adaptations in the menu were non-obtrusive. The statistics presented in the adaptive menu allowed participants to gain a better understanding of their app use behaviour in places. However, awareness alone did not provide motivation to change behaviour. Further steps will be required to encourage users to manage their app use behaviour in places. The app launch history of participants were collected in this user study to create a data set of app use in places that can be explored in the interactive visualisation.

The cost of app navigation menus was modelled and evaluated using the app launch dataset to demonstrate how this might find the optimal navigation mode given the context of the user in personal places.

The next chapter builds on the tools created thus far, to design a situated display that focuses on co-located social situations and the negotiation of smartphone notifications.

# Chapter 7

# Collaborative Media Sharing on a Situated Display

The previous chapter considered app tracking in personal places as a way to develop an adaptive homescreen that enables apps to be navigated quickly in places, and encourage self-reflection by displaying personal app use statistics on the homescreen. This chapter considers ways of supporting social engagement and the negotiation of smartphone notifications in places, by co-ordinating smartphone content on a situated display (RQ-3 of Section 3.1). Cast Together, a probe that demonstrates inclusive and unobtrusive mobile interactions with a situated display, is developed. The hedonic and pragmatic design of the Cast Together probe is validated in a social setting, and feedback from an informal user study evaluated in places leads to initial insights with the probe. To investigate the impact of notification display choice for negotiating smartphone interruptions, six notification displays are evaluated while performing a typing task. The results from these user studies are presented, with implications for the design of a situated display that shares smartphone content with co-located persons.

## 7.1 Inclusive and Unobtrusive Mobile Interaction with a Situated Display

There are opportunities to support co-located persons with inclusive and unobtrusive mobile interactions, by connecting smartphones to a shared display. ‘Inclusive’ interaction is defined to mean actions produce visible effects for spectators, and performers are granted equal opportunities to share. ‘Unobtrusive’ means manipulations are low attention, explicit gestures or implicit by passing control to an agent. Compared to positioning all mobile devices together [99, 134], it is argued that sharing media and events on a situated display

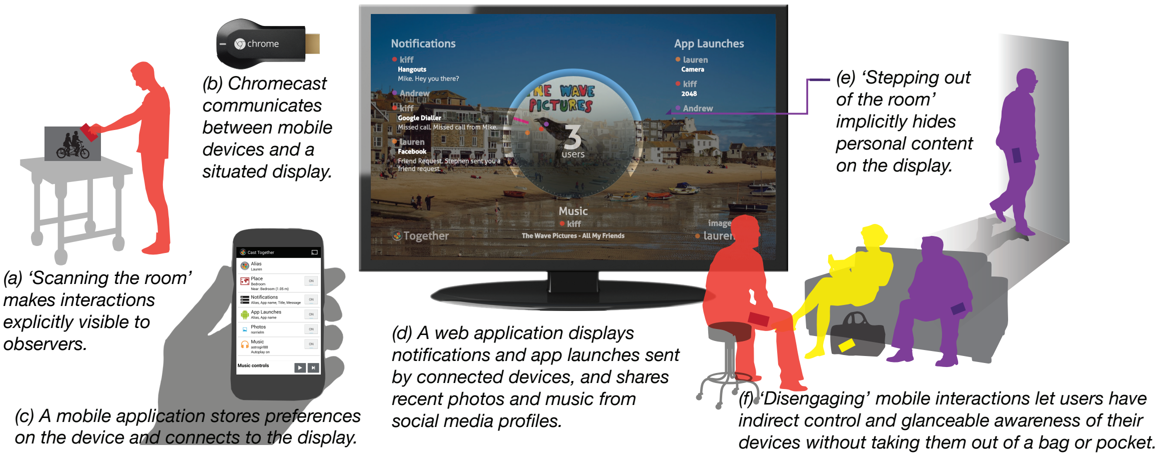



Figure 7.1: Cast Together is a probe for inclusive and unobtrusive mobile interactions.

moves the attention away from personal devices, and can encourage co-located persons to focus on social engagement.

Private smartphone interactions can negatively impact a social situation by making co-located persons feel excluded [19], and interruptions can draw multiple people away from a social situation to engage with a private display (‘collateral disruption’) [73]. In a shared media application, selections become chaotic when all users become actively engaged in selecting content on their personal device, instead of appreciating the selections of others [2]. [117] demonstrated that allowing users to vote on music choices creates an inclusive experience, since all users can become equally involved in the selection of music. However, requiring users to actively vote for songs requires a visual attention-switch to a private display, which might lead to distraction. Presenting historical data back to the user can encourage self-reflection [94], and can stimulate discussion with other people [19].

Cast Together is implemented as a probe to investigate the impact of inclusive and unobtrusive mobile interactions in a co-located environment. The selection of songs is automated by utilising the listening history of users that are stored in music applications or scrobbling platforms, such as Last.fm. Similarly, photos are selected from those uploaded to social media platforms, such as Flickr. Automating the selection of media demonstrates both an inclusive and unobtrusive user experience. Additionally, selecting publicly shared photos from personal social media profiles is demonstrated to stimulate conversation and encourage users to reflect on their digital identity. In the following sections, the design of Cast Together is related to existing work, and the results of a preliminary user study are shared.

### 7.1.1 Prototype Design: Cast Together

Google Chromecast, displayed in Figure 7.1 (b), is a popular low-cost solution for connecting a smartphone to an external display. Cast Together is implemented as custom sender and receiver applications for Chromecast, with screenshots shown in Figure 7.1 (c) and (d). The Android sender communicates via Wi-Fi with the Chromecast connected to the situated display. This app contains the cast icon that controls the display to join, and stores the following optional preferences for sharing content and controlling privacy: an alias to be associated with, a photo and music collection, and the level of detail to share about app launches and notifications.

#### Identity

To avoid the ambiguity of contending users [35], events and selections are associated with a colour and an alias. The alias identifies a device with app launch and notifications events, and the current music and photo selections on the situated display. The alias is displayed beside a unique colour that it is automatically assigned when the device connects to the display. A clock in the center indicates the number of connected users in the co-located group. The album art of the active music selection is displayed as the clock background.

#### App Launches and Notifications

The app name, title, and message of notifications can be shared at optional levels of detail to let users negotiate interruptions at a glance, and avoid an obtrusive visual attention switch [124] to a private screen when a notification is not important and can be ignored. If an app is launched on a personal device, a visual effect is created by sharing the alias and app name on the situated display. App launches and notifications display under the appropriate heading. If a user has chosen to exclude an app from the display, only the alias is revealed. The most recent event animates in from the top, and older events hide at the bottom. When a notification is dismissed or an app is closed on the smartphone, these events animate out entirely.

#### Photos and Music

Social media users maintain online identities by updating profiles of historical activity. Flickr and Last.fm are web services that let users upload photos or 'scrobble' music to profiles. Cast Together provides coarse-grained control over media collections by allowing users to present themselves through these public profiles. Photos display as the background by storing a Flickr user name, or a search term for a 'favourite thing'. A music playlist is generated by

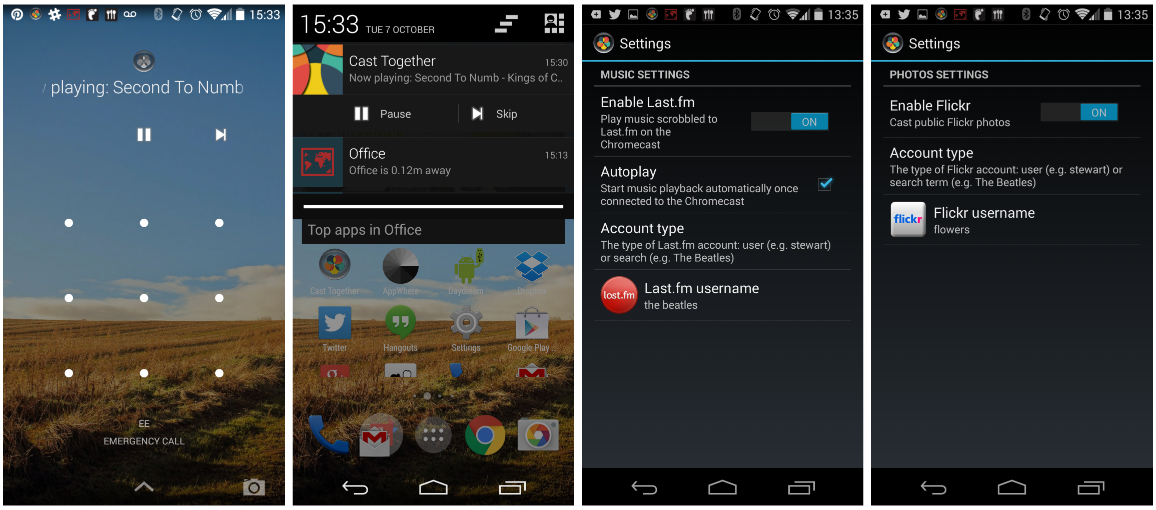



Figure 7.2: Photo and music preferences and controls.

entering a Last.fm user name or 'favourite artist'. The active song and photo rotates between users, and the alias associated with the current selection is indicated on the display. This automated round-robin schedule is an alternative policy to [117], which requires users to explicitly interact with a device and actively vote for songs. Photo and music collections are chosen from recent history, providing a discussion point when personal profiles are chosen, as these reflect the latest music listened to and photos shared by the user. Collections are shuffled to ensure that sessions are varied when profiles are not regularly updated. As fine-grained control is required for music playback, any connected device can change the volume, and play, pause or skip the music.

### Explicit Interaction

Privacy levels and preferences of music and photo profiles can be recorded to an NFC tag, and retrieved by explicitly holding the device close to the NFC tagged object. 'Scanning the room', illustrated in Figure 7.1 (a), supports the idea of coupling bits and atoms [83] by relating physical objects to digital profiles. For example, a favourite artist can be linked to a band poster. NFC also allows users to 'beam' preferences to each other by holding the devices back-to-back when the app is open, making the act of sharing a visible performance.

### Implicit Interaction

Sharing a full notification message is sometimes more appropriate in an intimate home environment than at work. To respect the user's privacy and reduce the need to explicitly update settings in different places, Cast Together adapts to the preferences shared with each display. Similar to proxemic interaction, events can be implicitly hidden when a user 'steps out of the room' with their device by detecting when a Low Energy Bluetooth beacon is out of range, as illustrated in Figure 7.1 (e). If a user ‘steps out of the room’, event details react to show only the alias, and animate back in when they return. Cast Together can also be instructed to automatically play music, or automatically connect to known displays when one is detected and idle, allowing the device to remain in a bag or pocket, as illustrated in Figure 7.1 (f).

#### Music Playback Limitations

Last.fm is a music scrobbling service. That is, when a Last.fm user listens to music with their subscription service, such as Google Play Music or Spotify, the artist and song title are recorded in their profile. This allows users to reflect on their music preferences, and the service can make users aware of upcoming events for their favourite artists. Last.fm radio realises its limited playback through YouTube, a video streaming service. However, Last.fm does not provide a platform for music playback itself.

At the time of writing, integrating Cast Together with a music platform that supports a subscription service, and enables search and casting to a Chromecast is not available. Given this constraint, music playback is temporarily realised through the YouTube video streaming service. For each song received from a Last.fm profile, the artist and title is searched for on YouTube. If both entities are contained in the top search result, the song is played. If only one of the song or artist entities match, the song is played, and a reference to the YouTube title is displayed beneath the Last.fm title on the display. If neither the song title or artist name match, the song is not played. As there is a risk that the songs listed in a Last.fm profile are too obscure or unavailable on the music service, it was decided that this user skips a turn, and a selection is made from the playlist of the next user in line. This avoids a long interruption to find an available song.

Given the commercial implications of playing music through YouTube, and the privacy and security implications of sharing smartphone events on a shared display, the evaluation of the Cast Together probe is limited to lab evaluations and trials with known users. Further development and research is required to make this prototype appropriate for use in the wild.

### 7.1.2 Evaluation: User Experience of Cast Together

Four groups of 2 participants were invited to experience the situated display in a social setting, with the aim to evaluate the hedonic and pragmatic design of Cast Together.

Figure 7.3: Experiment setup.

### Social Task

A board game was chosen as a social activity for participants to take part in while experiencing the situated display. Dixit[1] is a simple card game where players take turns to be the storyteller who says a word, phrase or description that resembles the picture on a card in their hand. Other players choose a card from their own hand and give it to the storyteller. Once all players have handed over a card, they are shuffled together and revealed. Players then vote to guess which card was played by the storyteller. Players gain points by correctly identifying the storyteller's card, and for each player who incorrectly guessed the card that they handed over instead. The storyteller gains points if $n$ players correctly identify his card, where $M$ is the total number of players and $0 < n < M$. The game requires 3 - 6 players. A games master was appointed to take part in all trials, but who did not contribute to the results. The partner of the experimenter was appointed as the games master, who was familiar with both Cast Together and the board game. The role of the experimenter was to observe social interactions with Cast Together.

Typically, the game ends when there are no cards left to fill a hand of 7 cards. However, to fit with the constraints of the short evaluation, the winner is identified as the player with the highest score after 15 minutes. During this game, there are periods to wait for the storyteller

[1]Dixit board game: `http://www.dixit.com`

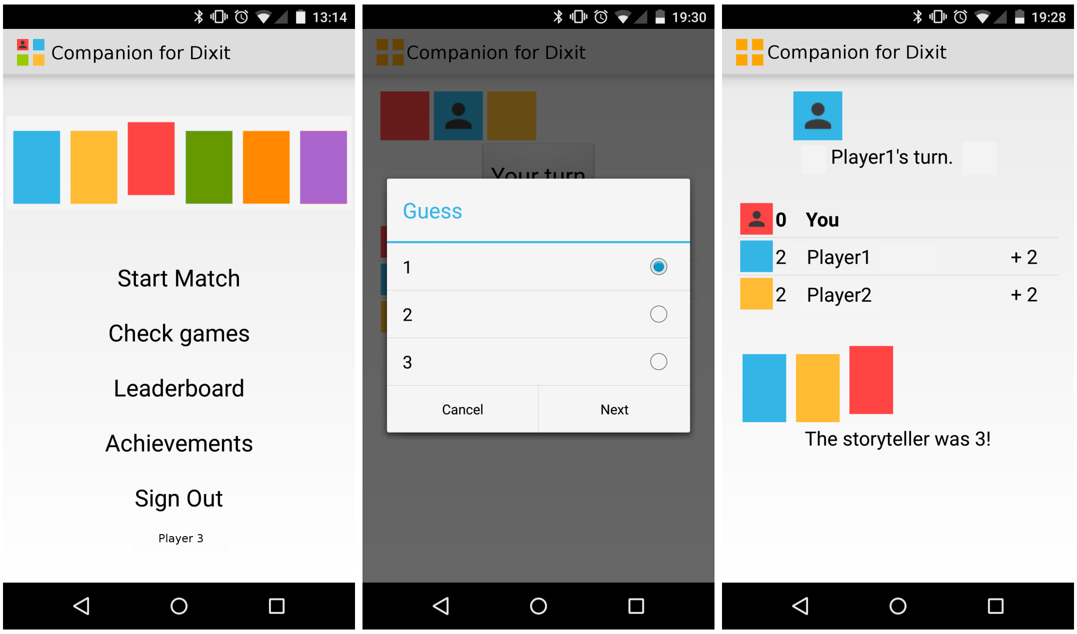



Figure 7.4: Companion app for the Dixit board game.

or other players to choose a card. Furthermore, players require inspiration to relate to the abstract images on the cards, and to think of a phrase that other players, in particular those who are not in the lead, will also understand. Therefore, Dixit is an appropriate game to engage participants in a social activity, while allowing time to experience the personalities projected through social profiles on Cast Together.

An element of the board game that can take time is calculating the scores. To simplify the scoring of the board game for the short experiment, an app was created to automate the physical counters and present the scores. The app replaces the physical counters of the board game by allowing co-located players to guess which card was added by the storyteller, and which card was the one that they added. After all players have voted, the app updates the score on all devices, and reveals which card was put down by the storyteller, as displayed in Figure 7.4. Requiring all participants to enter their guess via the app changed the game dynamic from a social discussion of the scores to each participant one at a time entering their scores into an app on the smartphone provided to them. Therefore, the app also acted a way for participants to experience app launches being shared on the situated display. In a test run with this app, a test player responded to a text message during the board game, while other players were waiting for him to place his vote: The other players used Cast Together to notice that he was not using the board game app, and asked him to continue playing the game. In the experiment, participants were provided with smartphones and so the temptation to perform personal app activities was not possible. However, this initial insight demonstrates one use case of the sharing of the app name on the situated display.

### Setup

The evaluation was set up in the living room environment of the experimenter, displayed in Figure 7.3.

### Equipment

The Cast Together sender app was installed on Nexus 5 smartphones that were provided for each participant. A Chromecast connected to a 28” television was positioned at the side of a table and in view by all players, as shown in Figure 7.3. To simulate the arrival of notifications during the short study, social media profiles were set up on each device, an approach taken by [100]. A Twitter account was signed in to one device, which received tweets from a Twitter account that reports facts (@QIElves).

### Participants

8 participants (2 x 4) took part in the evaluation. All participants were familiar with the board game, and were friends with at least one other participant prior to the experiment, which created a natural social setting. Three social groups participated as family (teacher, housing), friends (technical), and work colleagues (designers, Ph.D. students). All participants were social contacts of the games master, and were aged 20 - 40 and two were female.

### Measurements

The AttrakDiff questionnaire [75] was used to quantify the perceived beauty, goodness and usability of Cast Together. This questionnaire is available as Appendix F. Participants can indicate their perception of the product by rating 24 bipolar adjectives on a 7-point Likert scale. Four product dimensions are evaluated: Pragmatic Quality (PQ); two hedonic qualities, Stimulation (HQ-S) and Identity (HQ-I); and Attractiveness (ATT). PQ describes the usability of a product and indicates how successfully users are in achieving their goals using the product. HQ-S indicates to what extent the product can support an inherent need to develop and move forward, in terms of novel, interesting and stimulating functions, contents and interaction and presentation styles. HQ-I indicates to what extent the product allows the user to identify with it. ATT describes a global value of the product based on the quality perception. Hedonic and pragmatic qualities are independent of one another, and contribute equally to the rating of attractiveness. Three additional questions were provided to gather qualitative data about the impact of Cast Together on the social experience.

### Procedure

Each group was introduced to the features of the situated display by the experimenter, and were asked to enter their music and photo preferences in the Cast Together app on the smartphone provided to them (10 minutes). The game was then initiated by the experimenter, who helped participants join the game in the companion app. After 15 minutes, a timer notification alerted the experimenter that the game had ended. This notification was also shared on the situated display and informed the group. Finally, all participants were asked to complete the AttrakDiff questionnaire [75] (5 minutes).

## 7.1.3 Results

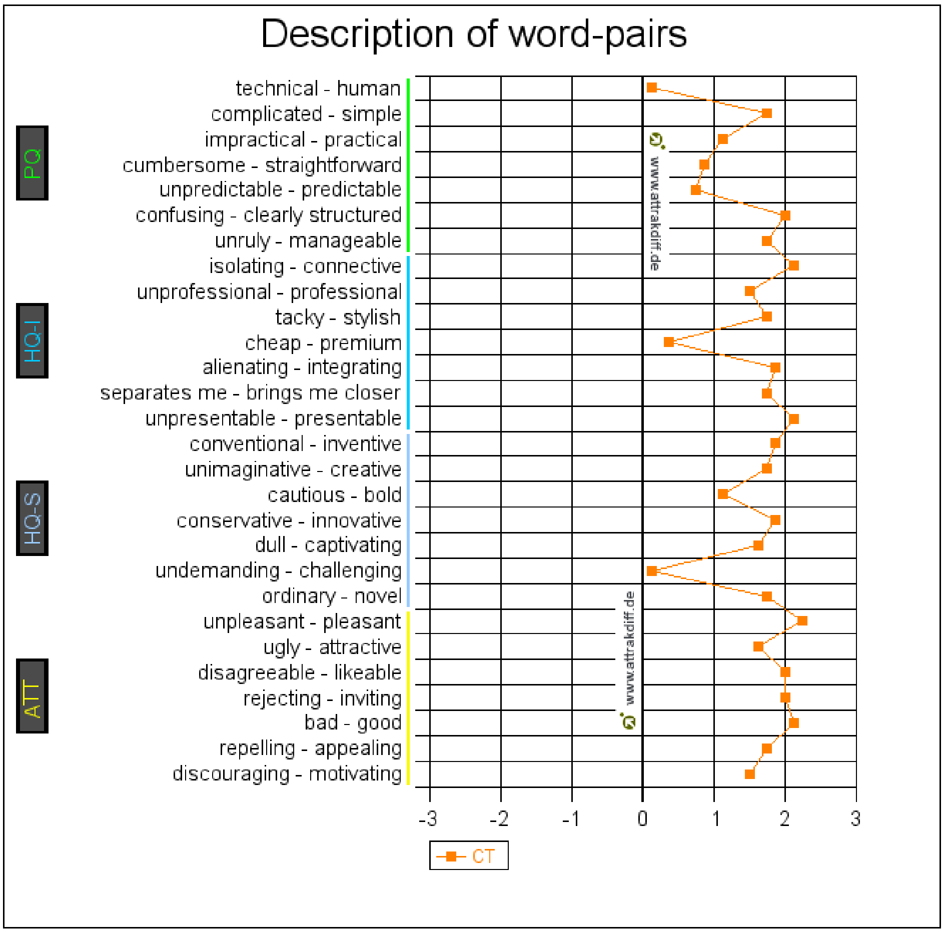
The results of the AttrakDiff questionnaire are reported, in addition to the subjective comments made by participants and observations made by the experimenter.

### AttrakDiff

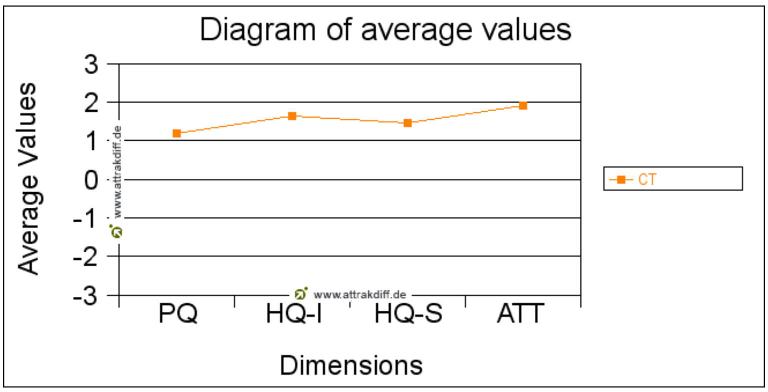
The mean values of the word pairs are presented in Figure 7.5 (a). Of particular interest are the extreme values. These show which characteristics are particularly critical or particularly well-resolved. Figure 7.6 displays the values of each word pair separated by participant, to show the variation in the results. Figure 7.5 (b) displays the average values of all four dimensions. Hedonic quality distinguishes between the aspects of stimulation (HQ-S) and identity (HQ-I). Furthermore the rating of attractiveness (ATT) is presented. The product's attractiveness value is located in the above-average region. The overall impression of the product is very attractive.

With regard to hedonic quality, the product identity is located in the above-average region. It provides the user with identification and thus binds the user to the product. In terms of identity aspects the product is classified optimal. In particular, three word pairs that describe the inclusive nature of the product were rated highly: 'Connective' (2.13, SD=0.83), 'Integrating' (1.88, SD=0.83) and 'Brings me closer' (1.75, SD=1.28).

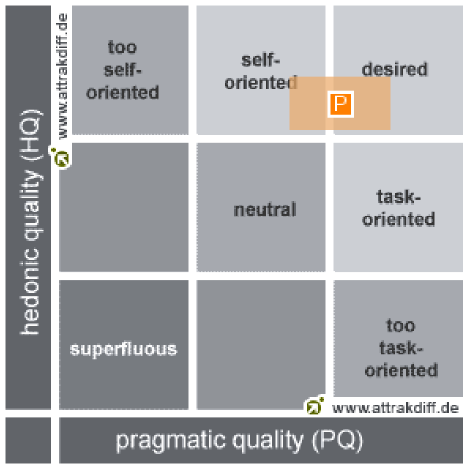
The stimulation of the product is located in the above-average region. The AttrakDiff report states that it meets ordinary standards, but to motivate, enthral and stimulate users even more intensely, Cast Together requires improvement. However, this result poses a mismatch between how AttrakDiff measures stimulation with a digital product compared to stimulation in the user environment. Cast Together is intended to be an ambient presence that stimulates a co-located group when desired. Therefore, this average result could indicate positive result for Cast Together. In particular, the rating 'Undemanding' (0.13, SD=1.13) was below-average, and 'Simple' (1.75, SD=1.39) and 'Manageable' (1.75, SD=1.39) were




(a) Word pair diagram.




(b) Average values.




(c) Overview of HQ and PQ. HQ is above average, in the 'desired' coordinate. PQ is bordering the 'self-oriented' co-ordinate.

Figure 7.5: AttrakDiff results with 8 participants.

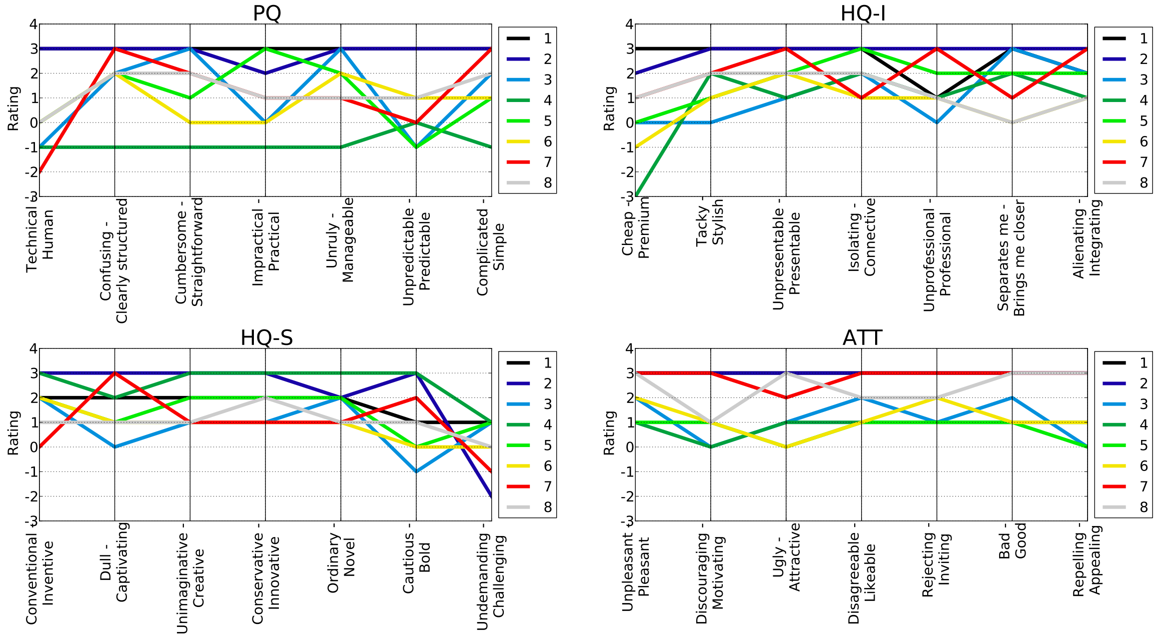



Figure 7.6: AttrakDiff word pairs separated by participant Id.

above average, which may indicate a negative experience with a product that is intended to actively engage its users.

An overview of the AttrakDiff dimensions are displayed in Figure 7.5 (c). The product's user interface was rated as 'rather desired', and was not clearly 'pragmatic' because the confidence interval overlaps into the neighbouring character zone. The user is assisted by the product. However, the value of PQ only reaches the average values. Consequently there is room for improvement in terms of usability. The confidence interval PQ is large. This could be attributed to limited sampling or to greatly differing product ratings.

## Observations

Participants mainly chose images of food, cities and animals as their favourite thing. As images appeared, players were observed to draw the group's attention to the display by making comments such as "who chose pancakes as their favourite thing?" [P5] or "that's a great idea for a cake!" [P2].

Favourite artists, or artists that participants shared that they were going to see live in the near future, were selected as the music profiles. Though some selections were obscure, participants were surprised when other players knew the artist that they had selected, uncovering a shared musical interest. With other selections, players groaned at more provocative selections that were made, and the group shared in the enjoyment of complaining about particular artists.

In addition to music and photos, some notifications appeared were noticed by participants.

During the game, P8 drew attention to a Twitter notification that appeared on the display by asking P7, whose alias was assigned to the event, if they had been tweeting. This comment indicates that the system did not have suitable accountability for this notification, as the P7 did not actively send the tweet himself. At the end of the experiment, P8 noticed a low battery notification on the display, and shared the advantage of the display being able to identify when the smartphone needs charged.

### Subjective Opinion

Participants reported on what they liked and did not like about using Cast Together while playing the board game.

All participants liked that "everyone was included in music choice" [P6], "good shared music" [P7] and that it was an "interesting concept to get people sharing music etc." [P5]. P1 "liked being involved in setting the atmosphere by choosing images and music. I liked the [board game] app also". Similarly, P2 liked that "everyone had input and could contribute equally. It generated discussion and added extra interest to the game and it was easier to keep track of the score". P3 liked the "amusing image results" and that "musical preferences start conversations". Similarly, P8 liked "the continued presence of Lego", highlighting that personalities can be projected through image search. Additionally, P4 liked "the exciting tension of what was going to happen next", which suggests that the neutral rating for the predictability word pair is a positive feature for Cast Together. Overall, participants felt that Cast Together "was a lot of fun and has lots of potential applications. It was very easy to use and inclusive" [P2], "It's great, very useful" [P7] and "I like the idea and where it's going" [P4].

Several limitations were identified in the comments. P3 felt that 'in small groups, the music can get repetitive' and there is a "lack of control of specific music" [P5]. Some participants felt that "it took some time to get everyone set up and connected but not too long" [P2] and P6 identified "connecting" as a feature that they disliked. Additionally, P5 felt that "I don't think its something I would use at parties, unless it was mainstream enough that everyone knows how to use it. Otherwise it would cause confusion".

P1 commented on the impact of Cast Together to the social situation: "The [board game] app we used meant we could pay closer attention to individual stats while not interrupting the conversational flow". However, "the game we played involved using knowledge of friends, so the choices of music or photos could at times exclude game options as they were then too obvious" [P1]. P1 had to adjust her game tactic while using Cast Together, as P2 selected his favourite animal as the background to Cast Together, making this common knowledge to the other players.

### 7.1.4 Discussion

Cast Together demonstrates inclusive and unobtrusive mobile interactions with a situated display. Cast Together coordinates media and events from devices present in a place automatically, and is designed to be a glanceable display to negotiate interruptions from smartphone notifications, and to stimulate conversation with media preferences. By reacting the content shared on the display to Bluetooth beacons and allowing profiles to be tagged to objects in a place, Cast Together acts as a probe for situated interactions. An evaluation with 8 participants validated the hedonic and pragmatic design of Cast Together, and the sharing of media was found to stimulate conversation without active engagement with a smartphone. The next step will be to evaluate Cast Together in more dynamic, social environments to understand the use cases of sharing smartphone events and self-presentation in co-located social situations.

## 7.2 User Study: Evaluating Cast Together in Places

Cast Together was implemented as a probe to investigate inclusive and unobtrusive mobile interaction with a situated display. To uncover the potential use cases of sharing smartphone events and presenting users with social media profiles in co-located places, it was desirable to conduct a longitudinal user study. However, given the privacy and security implications of sharing smartphone events on a shared display, and the commercial implications of playing music through YouTube, this probe is not yet ready for use in the wild. Therefore, the author limited the use of Cast Together to herself and close friends during a 7 month evaluation between 1 October 2014 - 1 April 2015. To provide a balanced account, the author reports on the feedback and observations that are supported by qualitative and quantitative evidence shared by her associates in questionnaires. Though the inclusion of the author, and relationship of the author to the participants introduces a bias to the results and is a limitation of this user study, these insights are a first step to identifying the impact of sharing smartphone events and using social media profiles to present personalities with co-located persons in personal places.

#### Participants

Three participants volunteered to evaluate Cast Together in their places: the author, her partner and her partner's brother, who for simplicity and anonymity will be referred to as Audrey (author), Paul (Audrey's partner) and Brian (Paul's brother). Paul and Brian are motion graphics artists who own a small business. They share an office together, where they can be found most days of the week. Paul lives in a small apartment with his partner Audrey, an interaction researcher and software engineer. When they are not busy with work, Paul and Audrey host parties and attend board games events with their friends, including Brian.

### Procedure

In the first week, participants were asked to connect to the situated display manually. The auto-connect feature was enabled in the second week, allowing the Cast Together application to launch on the Chromecast when the device is otherwise idle. In week 3, NFC tags were distributed to the participants, which could be used to bookmark the profiles and settings that Cast Together is configured with and enable situated interactions to load different profiles by tagging physical artefacts. In week 4, Bluetooth beacons were provided, enabling the content on the Cast Together screen to react to the presence of each participant near the shared display. After the initial 4 week period, participants were allowed to keep using Cast Together in places for 7 months.

### Equipment

Cast Together, Appwhere and My Places were installed on the Nexus 5 smartphones of all three participants. Two additional Nexus 5 smartphones and one Nexus 7 tablet were provided for guests to connect to Cast Together during social events. To discourage participants from using Cast Together in untrusted environments, Cast Together was only enabled on known Chromecast devices. Participants would then have to claim ownership of a television or monitor in a space to connect the Chromecast and launch Cast Together. The three Chromecast devices belonging to each participant were registered in the Chromecast Developer Settings, and one additional Chromecast was provided to be used in the office of Paul and Brian.

### Measurements

Qualitative feedback was received in questionnaires during the first four weeks of the evaluation, where each week a new part of the system was added. After 7 months of using Cast Together, participants were provided with questionnaire to record insights about the places where Cast Together was used. These questionnaires are available as Appendix G. Throughout the study, quantitative data was recorded on App Engine and Google Analytics about the duration of Cast Together sessions and number of smartphone events shared.

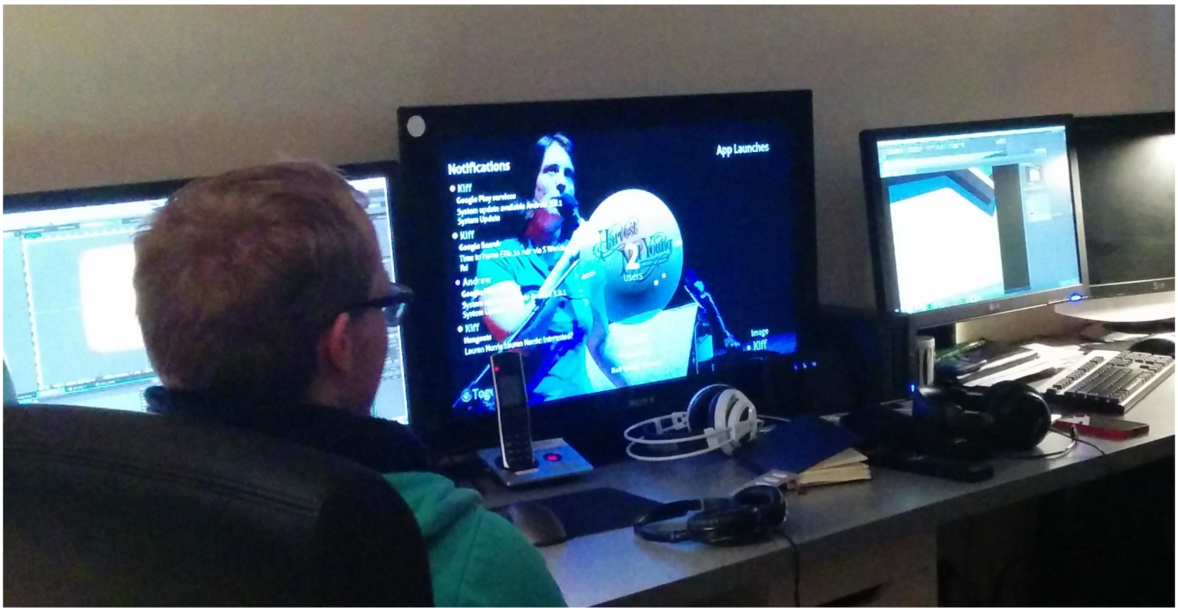



Figure 7.7: Cast Together was set up between two desks in a small office environment.

### 7.2.1 Results

941 Cast Together sessions were hosted during 1 October 2014 - 1 April 2015. Of the 148,143 events that were recorded, 48.6% of events were notifications, 22.0% app launches, 14.5% photos, 9.8% music. On average, 157.43 events were shared in a single Cast Together session. The results report on the findings in each place where Cast Together was reportedly used: an intimate office and home environment, a social party and board game event, and a family holiday occasion.

#### Office

Cast Together was used in an office shared by Paul and Brian. The display was set up between the two desks, as displayed in Figure 7.7. App activity was monitored, and it was found that Cast Together was launched in the office 119 times by Paul and 172 times by Brian. At the end of each week, participants responded to questionnaires about their experience. Sharing notifications on the display was "good for when we both have the same ringtone and it's ringing in one of our pockets and we don't know whose" [Paul], and "makes me more likely to get stuck in work when I'm on a streak" [Brian]. This motivates the sharing of notifications to focus on a task, and to avoid collateral disruption when alert tones are similar. Brian felt that sharing music "was great as it stops you from listening to the same stuff" and "the nice slideshow effect meant I could display a bunch of images for inspiration for projects I am working on". Connecting automatically "was very useful as forgetting to turn it on was not an issue" [Brian]. However, "Cast Together came up with a few images I wanted to use/save, but since it was just a slideshow I could not". Paul positioned the NFC tags "mainly around


Figure 7.8: Home: NFC tags in the corner of the picture frame link to photos and music profiles .

my desk or up on the wall to the left of me where all drawings/inspiration materials go" and were "very handy which made it easier to change the music and images I wanted. This gave me more time to focus on my work instead of getting distracted and browsing the internet after I have picked some music".

### Home

Cast Together was used for 7 months in the home of Paul and Audrey. At home, Paul found "the display is useful when making dinner and generally when doing housework... I can listen to music while doing dishes and preparing food". Paul likes that 'I rarely have to think about what music to play as the display will play indefinitely' and he finds it "interesting to hear the music my partner has been listening throughout the day".

Paul also found the notifications to be useful while cooking: "I also regularly have a timer going on mine or my partner's phone which is monitoring cooking times. It works very well on the display as the timer periodically displays on the screen as a notification, so I can check cooking times without having to consult my phone. This is especially helpful while cooking and cleaning, as often I will be checking notifications while my hands are not clean/dry" [Paul]. In addition to the timer notification, "other notifications can also be helpful to see on the display while at home as I can quickly distinguish if they are important (e.g. phone call vs Twitter/Pinterest etc)" [Paul]. When asked about the privacy of notifications, Paul responded, "As this place is in the home I share with my partner, I don't have any problem with sharing my notifications or app usage in this scenario".

Paul finds that "the biggest limitation with the display for me is when it doesn't auto connect. I find it easy to forget to connect as a user, and then end up looking at my phone more or missing notifications". This was especially frustrating when "launching another app through

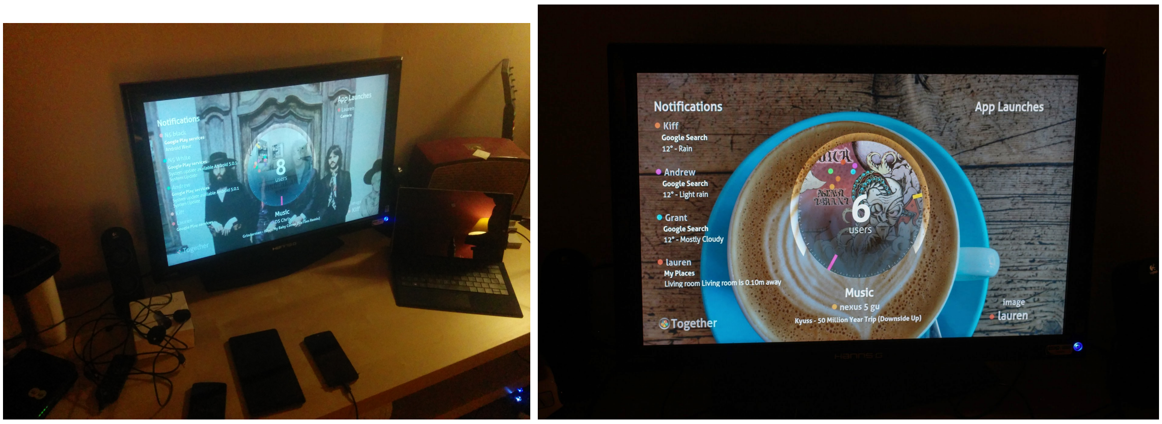



Figure 7.9: Party: Devices were provided for guests who did not have an Android smartphone.

the Chromecast, which overrides Cast Together. It can seem a little tedious to have to re-launch or join Cast Together, so I often find myself not getting round to it. I'll sometimes only notice I have not signed in when I realise I've been listening to several tracks in a row of my partner's music" [Paul].

### Party

Paul and Audrey used Cast Together to play music and display themed photos at two dinner parties in their home. 11 guests attended a Halloween party, and 8 guests attended a traditional supper in celebration of the Scottish poet Robert Burns, including Brian who attended both events.

Paul "found the display works very well for a party. It makes it very easy to theme music and pictures towards a specific event. For example, at Halloween we instantly had pictures and music appropriate for the event". The zombie themed party was enhanced by performing an image search for zombies, and the Burns supper event was themed with photos of tartan and haggis. "Some of our guests were able to install the app and take part as a user, we also had other devices signed in to allow non-Android users to select music and images. It makes for a fair and interesting Jukebox style set up for the music" [Paul]. During both parties, guests were interested in using their own Android 4.2+ devices to set their music and photo preferences with Cast Together. Audrey invited guests who requested access to the private Google+ Community, where they could become beta testers and download the Cast Together app. Guests who installed the prototype on their personal devices were instructed on how to remove the application before they left.

On one occasion, only allowing profiles to generate a music playlist was found to be a limitation. "During the party, two people requested a specific song, which is a limitation of the display, which will only allow you to choose an artist or will randomise a Last.fm history.

For most occasions, this works well, but this scenario showed the need to sometimes be more specific about music choice. We resorted to signing into my Google Play Music account and casting that for a while" [Paul]. Playing these songs on Paul's smartphone had further unexpected effects. "This was a good compromise and solved the problem during the party, but had the undesired effect of scrobbling these tracks to my Last.fm account, which in turn showed up the next time I connected to the display after the party, when I wasn't really in the mood to be playing these specific songs. I was able to skip past those songs, but eventually decided to delete them from my last.fm history to avoid having them appear in the future" [Paul].

Paul "noticed notifications on the display and had notifications of mine pointed out to me on a couple of occasions, which was helpful as they were messages and calls from guests, to say when they would be arriving etc. It might have been easy to miss one of them without the display in this scenario, as I may not have heard/felt my phone in my pocket while I was talking to guests with music on etc.". When asked about the privacy settings in the party, Paul responded, "I don't remember anyone having a problem with notifications being displayed on the screen. I didn't feel the need to reduce the detail of my notifications either, as I couldn't think of any notifications that I might want to be kept private".

### Holiday

Paul and Audrey set up the Chromecast during a one week holiday visiting relatives. The Chromecast was plugged into the television in the living room of the holiday home that was shared by Audrey's parents, sister, brother-in-law and nephew.

Audrey noted observations of the family learning how to share photos on the Chromecast on the first evening. Audrey's mother was using her smartphone to show her recent photos to the group. Audrey demonstrated the cast button that shared the photo on the television. Her father and sister asked if their devices could share photos on the big screen. They had opened their photos in an app that did not support casting, and were shown to the app that the mother was using. Though the 3 performers enjoyed sharing their photos from the device, the chaotic experience that followed was difficult to enjoy as the remaining 3 spectators, as storytelling was frequently interrupted by new photos being shared. During the free for all, a video was interrupted by a photo that someone had found to share, and Audrey's father did not realise that everyone could still see his photos as he browsed photos of his work.

Paul "found Cast Together to be most useful when the relatives were visiting, as they took notice and regularly commented on the pictures that were displaying on the screen. The pictures were a range of events and holidays that we had been to in the past, and even some pictures that we had taken on the current trip. It seemed like a nice organic way to view pictures together on a big screen... I'll often wonder how interested people really are in

Figure 7.10: Board Game Event: Cast Together was set up beside the table where a board game was played.

looking at every single photo you have to show from an event or holiday. Cast Together seems to allow for moments of engagement which will often spark a conversation". One of the Flickr profiles that was shared was Audrey's brother who was not attending the trip this year. Audrey's mother pointed to the display as a photo of her son and his wife appeared from their recent trip to America, prompting an update on how they had been since the previous year when they had visited. A search was also performed to aid a conversation about the Kelpies, which are structures in Falkirk that opened that year, which the family would be visiting the following month. Audrey's father pointed at an interesting photo of the Falkirk Wheel that appeared, helping his explanation of how it operated.

A limitation of the display in this scenario was that "we didn't have everyone that was staying in the house set up" and "it might have been useful if we did, as it turned out that several people on the trip had the same message alert tone which caused confusion on quite a few occasions. Had we all been connected, we might have been able to more easily see who's phone was requiring attention, or if indeed it was important enough to be such a big distraction" [Paul].

### Board Game Event

Brian invited Paul and Audrey to a board game event with 3 other guests in his living room. Brian asked if he could use Cast Together on his home Chromecast at the beginning of the event, which Audrey set up. Paul explained, "Cast Together allowed us to listen to music based on our history and search criteria, which made it easy to have background music throughout the night".

During the event, "songs sparked conversations and it was helpful to see whose track it was when talking about it" [Paul]. A song that Paul had scrobbled was one that he had learned to play on the piano. A guest told this to Brian's piano teacher, who did not know that Paul also played. "People did also take notice of and comment on different photos that appeared. It worked as a nice point of reference or subject of conversation, especially in this scenario, entertaining us when there was a lull in the game without being too intrusive" [Paul]. An example was photos from Audrey and Paul's recent trip to the Kelpies, and other guests were curious about what they were.

After the board game had ended, one guest was interested in the app that Brian had used to play music and asked how he could install the app. Brian explained the design of Cast Together, and that he and Paul had been using it in their office to manage their smartphone notifications. A notification that was displaying at the time was Google Now informing Brian about the time to reach the motorway, and everyone joked about where Google Now was planning to take him. While everyone was looking at the display, "the clock acted as a way of mutually noticing that it was probably time to call it a night" [Paul].

Paul shared that "one improvement on the evening would have been if everyone were connected to the display. As it was, three people present were not connected to the display, one of whom held up the game while focusing their attention on their phone. It would have been interesting to see how the rest of us might have acted differently if we were able to see what action they were performing on the phone. For example, had it been Facebook, we may have made a joke in order to 'shame' them into putting down their phone. As we had no idea how important the interaction with the phone may be, it seemed impolite to comment".

### Summary of Results

These initial results with Cast Together indicate several benefits of inclusive and unobtrusive mobile interaction in co-located environments:

1. Sharing notifications can avoid collateral disruption when alert tones are similar and can draw attention to important notifications.
2. Searching for profiles can lead to serendipitous discovery of music and images.
3. Sharing personal profiles lets others learn about previous activity and can spark conversations.
4. Implicit and explicit interactions can manage the sharing of content with little engagement with a personal device.

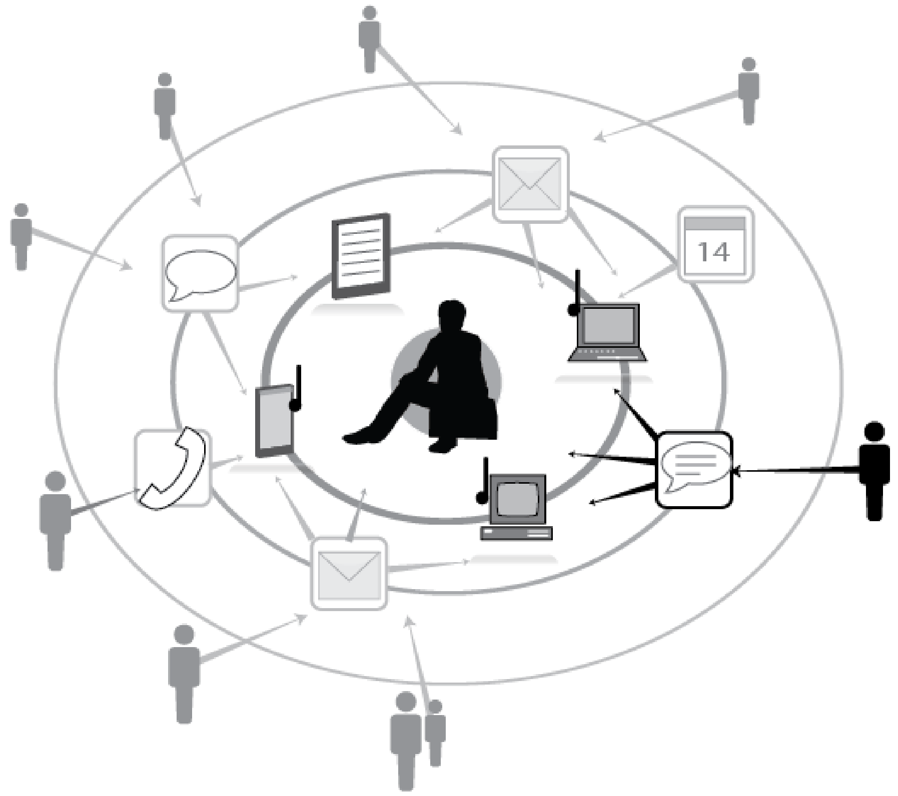


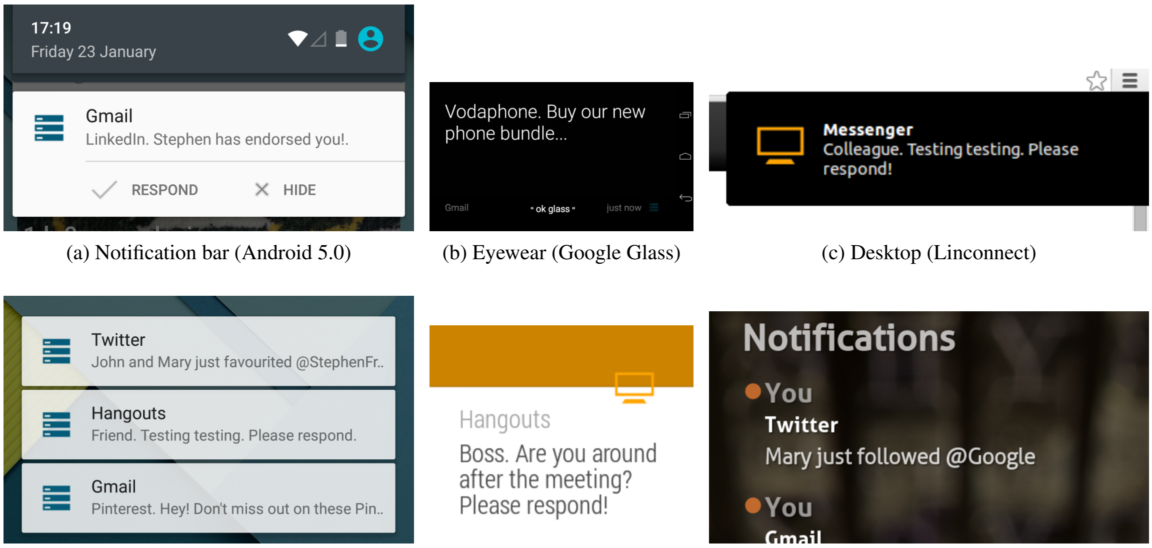

Figure 7.11: A user with many devices finds it difficult to manage all tasks at once.

### 7.2.2 Discussion

A user study was conducted with three participants who used Cast Together in dynamic social settings. The inclusion of the author, and the relationship of the author to the participants, are major limitations of this user study. Furthermore, this user group were comfortable with sharing media and events from their smartphones with their close family and friends, and so the privacy levels that were in place were not fully evaluated. However, these results are included as a first step to identifying the potential uses of Cast Together in places. A major benefit of Cast Together was that notifications could be read at a glance without interacting with the smartphone. The next step will be to compare the cost of reading notifications on a situated display to other notification displays when focusing on a task.

## 7.3 Choice in Notification Displays

Smartphone notifications provide awareness of important emails and messages. However, without consideration of the user and their context in the physical environment, notifications can be distracting, and frequent interruptions can result in stress [166]. Negotiated interruptions [106] let the user decide the onset of an interruption, and is an approach to managing notifications that can improve concentration on a primary task.




(a) Notification bar (Android 5.0) (b) Eyewear (Google Glass) (c) Desktop (Linconnect)

(d) Lockscreen (Android 5.0) (e) Smartwatch (Moto 360) (f) Situated display (Cast Together)

Figure 7.12: Notification displays.

On an Android smartphone, users negotiate interruptions via the notification bar. The notification bar displays a list of recent notifications in a pull-down menu, and is always one step away when the device is in use. However, when the smartphone is not in use, the notification bar can require many actions to read the notification. Displaying notifications on the lockscreen can reduce the cost of reading a notification when the device is in hand but not unlocked. Compared to the notification bar, the cost of responding to the notification is increased, as the device has still to be unlocked. By making it simpler to consume notifications, but more difficult to act, notification display choice has the potential to encourage the user to focus on a task, and be less likely to engage in prolonged smartphone habits when attention is committed to a task, whether a co-located social situation or an individual typing task.

External displays create new opportunities to deliver notifications to the user. Smartwatches and smart eyewear allow users to read notifications when the device is not in hand, by looking towards the wrist or glancing upwards. Notifications can also be displayed on a monitor as desktop pop-ups, or on a situated display in the user environment. As new ways of reading smartphone notifications become available, it is important to consider the impact that they will have on attention to everyday tasks.

This work contributes a study of six smartphone notification displays, shown in Figure 7.12, and their impact on attention to a typing task. Though many notifications do not require a response or can be ignored [132, 120], an equal number is selected to be ignored and acted on, in order to compare the relative importance of each notification type on subjective opinion, performance and resumption lag.

### 7.3.1 Research Methods: Measuring Interruptions a Typing Task

When a notification requires a response, it is possible to record the response event on the smartphone. However, if a notification is ignored, then there is uncertainty about when the user reads the notification. Additionally, if the notification can be read at a glance with a negligible delay, then there is uncertainty in the resumption lag. To improve the identification of when a participant reads a notification, face tracking with the Microsoft Kinect was explored as a way to record head movements towards each notification display. The Virtual Sensors tool from Section 4.1 integrates with the Kinect to record the position of the eyes, nose and mouth, and the 3D rotation of the face. Smartphone events were also recorded with this tool, including the accelerometer, notification events, and screen on and off events. However, the Kinect tracking was unreliable, and it was not possible to record the data for all participants. Face tracking was most unreliable for participants with hair that overlapped the face, and for participants who leaned out of view while looking towards a notification display. Therefore, the quantitative results are reported as a worst-case estimate of the interruption to the typing task.

### 7.3.2 Research Design: Notification Display Choice in a Typing Task

30 participants took part in a 1.5 hour lab experiment, and were asked to manage notifications while performing six 10 minute typing tasks. A £10 reward was provided for participation. The experiment setup can be seen in Figure 7.13.

#### Research Hypotheses

1. Glanceable displays will increase productivity more than smartphone displays.
2. Smartphone displays will increase the overall resumption lag compared to glanceable displays.
3. Notifications situated around the typing task will be preferred most overall.

#### Participants

Participants were aged 18 - 35 (mean=25, st.dev.=5), 10 were female. 26/30 participants studied Computing Science and were recruited from an undergraduate mailing list.

A preliminary questionnaire was provided to understand any source of bias. On a 7-point Likert scale, all participants reported to type on a keyboard regularly (6.63 +/- 0.82), and

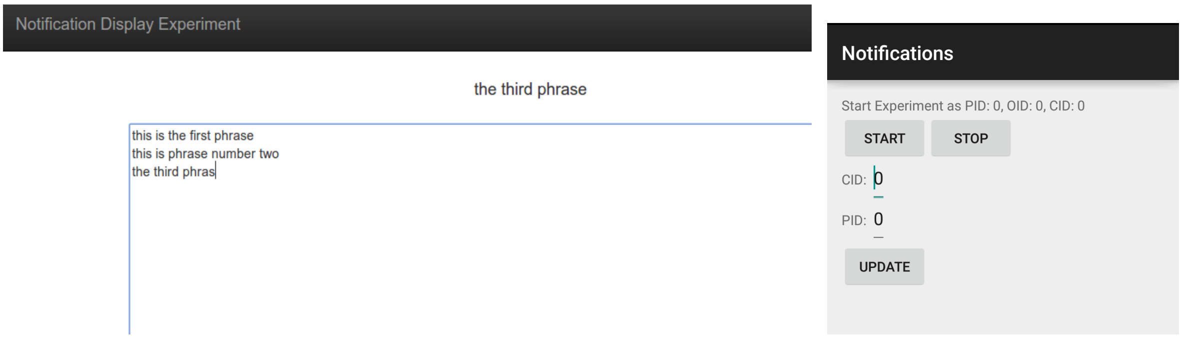
(a) Experiment setup

(b) Google glass

(c) Moto 360

Figure 7.13: Typing experiment setup. A test participant is negotiating interruptions with the situated display. (The photo of Google Glass is available with a Creative Commons licence.[2])

were more likely to receive smartphone notifications (6.28 +/- 0.91) than to respond (5.25 +/- 1.30). Participants were most likely to position the smartphone beside them (left or right) while typing at a computer (27/30), which is in line with the design of the experiment. Allowing participants to choose the onset of an interruption fits with the design of negotiated interruptions: 16/30 participants would finish a task before checking notifications, and 12/30 participants would allow notifications to interrupt their task. Few participants would wait until a predefined time of day. Participants were asked which display they would choose to read smartphone notifications while typing at a desktop PC. 16/30 participants indicated that they would choose a smartphone display, and 9/30 chose a desktop pop-up or situated




(a) Text editor. (b) Notification scheduler.

Figure 7.14: [Left] Typing task on Desktop PC. [Right] Notification Scheduler.

display, and only 4/30 chose a wearable.

### Typing Task

A custom text editor was implemented as a web page to log key strokes as participants typed phrases during the typing task, as displayed in Figure 7.14 (a). Phrases were consistently displayed in a single line at the top of the page to minimise head movements during the typing task. The next phrase appeared when a newline was entered in the editor.

### Notification Task

Smartphone notifications acted as distractions from the typing task. An app was developed to schedule notifications stored in a text file, as displayed in Figure 7.14 (b). At the start of each condition, a log file was created to record the timing of notification events. Messenger notifications were chosen as the response task, where the message requesting a response was prefixed with 'Please respond!'. In reality, smartphone users decide themselves whether a notification requires a response. Though it is possible to provide limited input to a smartwatch or smart eyewear, it is assumed that smartphone users take action on their smartphone, an assumption also made by [100]. Therefore, the response task always required participants to click the 'Respond' button on the notification bar, as shown in Figure 7.12 (a).

### Equipment

The equipment used in this experiment is displayed in Figure 7.13. A Nexus 5 smartphone running Android 5.0 was used by all participants. The notification scheduler app was installed, and notifications from this app were prioritised to ensure that no other notifications

[2] `https://commons.wikimedia.org/wiki/File:Google_Glass_Front.jpg`

would display during the experiment. The default notification bar and lockscreen were used as notification displays.

A Sony Vaio laptop running Ubuntu 12.04 OS was connected to a 12” monitor, USB keyboard and mouse, which acted as the desktop PC. The typing application was opened on the laptop in the Chrome web browser.

Linconnect,[3] an open source notification server for Android smartphones, was chosen to display desktop pop-up notifications. As opposed to services like PushBullet which display notifications in the web browser, Linconnect integrates with system notifications, which can be customised with NotifyOSD, including the size, position, timeout and colour. Notifications were customised to appear for 10 seconds in the top-right corner, as in Figure 7.12 (d). The Linconnect server was installed on the laptop, and the client was installed on the smartphone, and both communicated via a shared Wi-Fi connection.

The Cast Together application for Google Chromecast was used as the situated display, as described in Section 7.1. A Chromecast connected to a 18” monitor was positioned to the right of the desktop monitor, as shown in Figure 7.13. This Android application was installed on the smartphone to detect the notifications and app launch events, and communicated with the Chromecast through a shared Wi-Fi connection. A pre-selected collection of photos displayed in the background, and notifications appeared in the top-left side, as displayed as in Figure 7.12 (c).

A Moto 360 smartwatch and Google Glass smart eyewear were used as the wearable conditions, with screenshots displayed in Figure 7.12 (e) and (f). The Android Wear and My Glass applications were installed on the smartphone to pair with the devices via Bluetooth. Ambient mode was enabled on the smartwatch, which acts as an e-ink display when the screen is dimmed. The head-up trigger for Google Glass was found to be the simplest way to display notifications by rotating the head upwards after an auditory alert arrived on the headset. A head angle of $10\,^{\circ}$ was used as a sensitive trigger, and was tested with each participant before starting the smart eyewear condition.

A Microsoft Surface Pro installed with the Virtual Sensors evaluation tool was used to perform face tracking with the Microsoft Kinect v2.0. The Kinect was positioned below the monitor, approximately 20cm above the keyboard and 60cm from the end of the desk. Data from the Kinect is recorded to an SQL database with the evaluation tool, which can replay the experiment conditions. After the experiment, Kinect and smartphone data could be exported to a CSV format.

[3] Linconnect client/server. `https://github.com/hauckwill/linconnect-server`

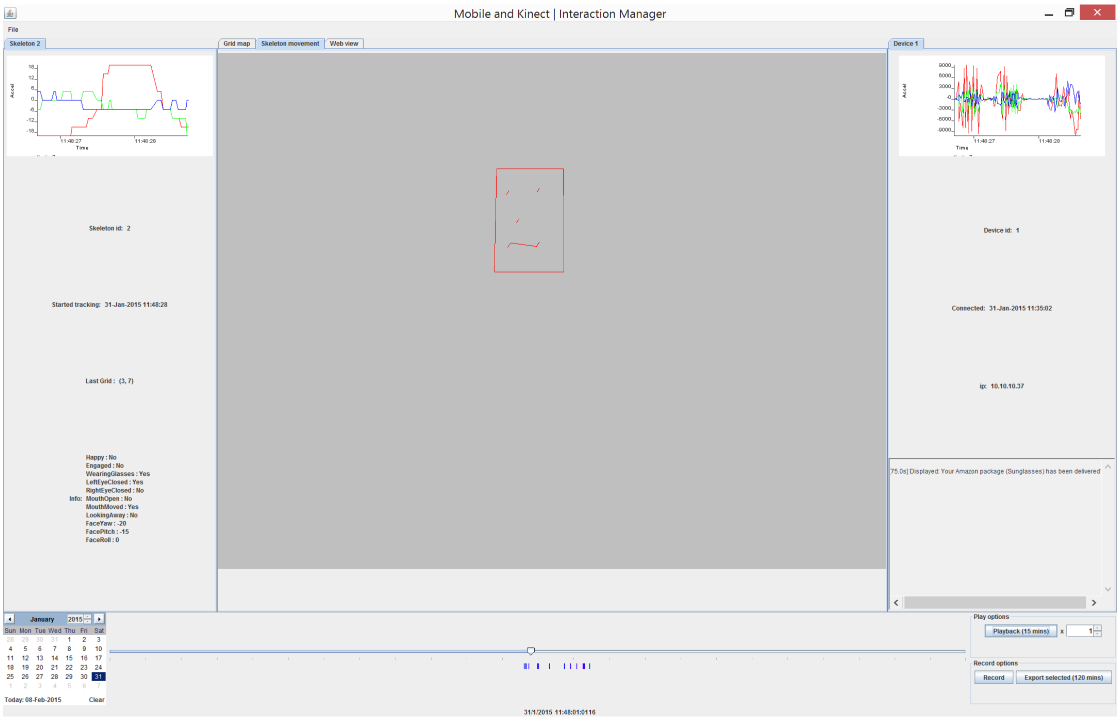



Figure 7.15: The Mobile and Kinect Interaction Manager recorded data from the facial position in relation to the notification displays, in addition to the events from the smartphone. The experimenter could monitor the smartphone events in real time to ensure that each participant understood which notifications to respond to during the training task.

### Procedure

At the start of the experiment, a consent form and preliminary questionnaire were presented. A short training task introduced the participant to the typing task and notification task. Next, the participant was asked to read the objective of the experiment, and clarified which notifications to respond to and which to ignore. The objective informed participants to respond to notifications in under 15s, and to resume typing quickly. A £20 incentive was awarded to the participant who responded accurately and who typed the most phrases.

At the start of each condition, a screenshot and short description introduced the notification display, which was set up by the experimenter. When the participant was ready, the typing task was started on the desktop, and the notification scheduler was started on the smartphone. The smartphone was positioned 15cm to the right of the keyboard.

The steps to a notification event are illustrated in Figure 7.16. (1) First, the user starts typing, and (2) a notification arrives on the smartphone, with an audio alert and a vibration. (3) Then the user decides to stop typing, and the notification is read on the appropriate notification display. (4) If required, a response is given from the notification bar of the smartphone. (5) Finally, the user resumes the typing task.

After 10 minutes, the typing task ended and the user was alerted on the desktop. The file

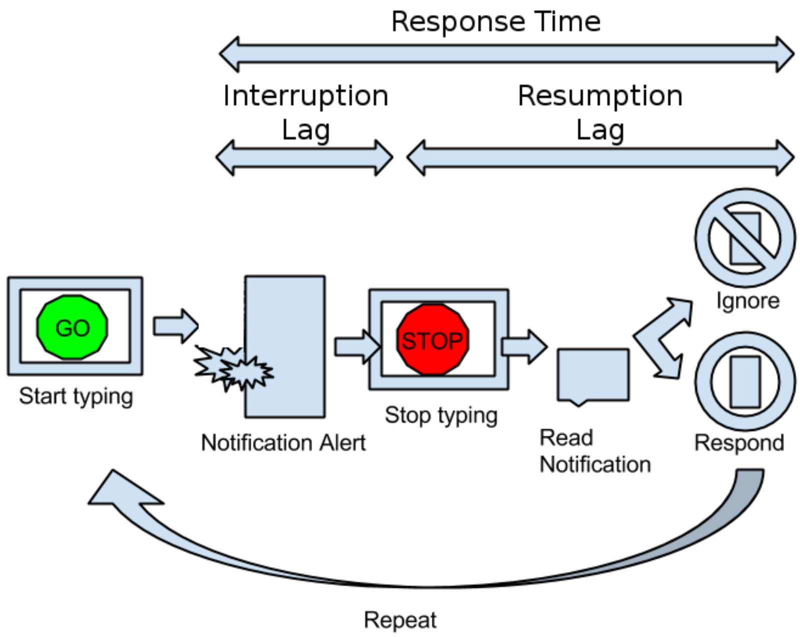



Figure 7.16: Flow of attending to a notification alert while performing a typing task.

generated by the typing task was downloaded, and the participant was asked to fill out a short questionnaire. After all six conditions were completed, the participant was presented with a questionnaire to rank all notification displays in order of preference. The questionnaires used in this study are available as Appendix H.

### 7.3.3 Statistical Design

This section explains how dependent variables were measured in the experiment, and how independent variables were controlled by selecting phrases to type and notifications to manage in each condition.

#### Task Dataset Selection

Two popular phrase sets for text entry [102, 154] were merged to create a large collection of simple phrases. Phrases were localised for UK English and phrases with names and numbers were removed. The remaining phrases were randomised and formed six groups of 80 phrases.

In an initial test with 2 participants, 25 seconds was found experimentally to be the minimum time required to respond to a notification and resume typing without feeling overloaded. Both test participants found that the notification bar was the most demanding, as it required accessing the notification bar even when a notification could be ignored. With an interval of 25 seconds, 23 notification events were issued in a 10 minute typing task. The first 3 notifications of each condition were not considered in the results, and the remaining 10 notifications could be ignored and 10 required a response. Six sets of 23 notifications based on real smartphone events were generated for the study, with examples displayed in each notification display in Figure 7.12.

### Task Measurements

All keystrokes were recorded with a timestamp in the text editor, which was downloaded as a log file at the end of each condition. From the final text entered into the text editor, counts were recorded of the number of phrases per minute (PPM), words per minute (WPM), characters typed (CPM) and keypress events (KPM). The average characters per minute was measured as the final number of characters, minus the number of characters typed during the test phase.

Notification events were recorded with the notification scheduler app on the Android smartphone, and were stored in a log file at the end of each condition. Timestamps were recorded when the notification was displayed, the Respond button was clicked in the notification bar, and after screen on and screen off events. Computed from these events were: the time between a notification alert arriving and stopping the typing task to read the notification ($t_{interruption}$), the time between stopping the typing task and returning to the typing task ($t_{resumption}$) and the time between a notification alert arriving and a response given to the notification ($t_{responded}$).

The Virtual Sensors tool described in Section 4.1 was used to record the position of the eyes, nose and mouth, and the 3D rotation of the face. Smartphone events are also recorded with this tool, including the accelerometer, notification events, and screen on and off events. As Kinect tracking was found to be unreliable, the resumption lag is reported as a worst-case estimate, and focus mainly on the subjective results.

After each condition, subjective responses were recorded in a questionnaire, and conditions were ranked in order of preference in a final questionnaire.

### Statistical Methods

An ANOVA was performed to test the significance of typing delays between smartphone vs. wearable vs. external displays, with $p \leq 0.05$. A log transformation was used to control for non-normal distributions in this timed data. A two-tailed Mann-Whitney U-test was used to test ordinal data, with $p < 0.05$ and a normal distribution to calculate U for 30 participants, with $z \geq 1.96$.

| | | Smartphone | | External | | Wearable | | Smartphone vs Glanceable | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Notification bar$^\diamond$ | Lockscreen$^\diamond$ | Situated display$^\dagger$ | Desktop pop-up$^\dagger$ | Smart Eyewear$^\dagger$ | Smartwatch$^\dagger$ | Smartphone$^\diamond$ | Glanceable$^\dagger$ | *Sig?* |
| Steps | Read $\downarrow$ | 7 | 5 | 4 | 2 | 5 | 5 | 6 | 4 | |
| | Respond (extra) $\downarrow$ | 1 | 4 | 8 | 8 | 8 | 8 | 2.5 | 8 | |
| | Avg. steps $\downarrow$ | 7.5 | 7 | 8 | 6 | 9 | 9 | 7.25 | 8 | |
| Prod. | WPM $\uparrow$ | 42.8 | 41.7 | 42.9 | 44.5 | 39.7 | 42.9 | 42.3 | 42.5 | |
| | CPM $\uparrow$ | 220.1 | 213.8 | 224.6 | 227.6 | 207.2 | 223.2 | 217.0 | 220.7 | |
| | KPM $\uparrow$ | 244.8 | 239.0 | 251.7 | 258.0 | 232.4 | 251.2 | 241.9 | 248.3 | |
| R. Lag | Ignore delay (s) $\downarrow$ | 5.61 | 4.53 | 2.41 | 1.86 | 4.12 | 2.41 | 5.07 | 2.72 | * |
| | Respond delay (s) $\downarrow$ | 4.39 | 5.58 | 4.80 | 4.45 | 6.75 | 5.38 | 5.34 | 4.99 | |
| | Avg. delay (s) $\downarrow$ | 5.00 | 5.05 | 3.60 | 3.14 | 5.36 | 3.84 | 5.03 | 3.99 | * |
| Subjective | Ease of Intepreting $\uparrow$ | 5.0 | 5.0 | 7.0 | 7.0 | 5.0 | 6.0 | 5.0 | 6.0 | * |
| | Ease of Concentration $\uparrow$ | 4.0 | 5.0 | 6.0 | 6.0 | 5.0 | 6.0 | 4.0 | 6.0 | * |
| | Convenience $\uparrow$ | 3.0 | 5.0 | 6.0 | 7.0 | 3.0 | 6.0 | 4.5 | 6.0 | * |
| | Frustration $\downarrow$ | 4.0 | 3.0 | 2.0 | 2.0 | 5.0 | 2.0 | 4.0 | 2.0 | * |
| | Urgency $\downarrow$ | 5.0 | 5.0 | 4.0 | 4.5 | 4.0 | 4.0 | 5.0 | 4.0 | |
| | Overall Preference $\downarrow$ | 5 | 4 | 3 | 2 | 5 | 2 | 5.0 | 3.0 | * |

Table 7.1: Summary of experiment conditions, with the mean of measured data and median of subjective data. $\downarrow$ indicates that a lower value is better.

### 7.3.4 Results

Table 7.1 provides a summary of results. The impact of notification display choice on productivity, resumption lag, subjective opinion and overall preference are reported in the following sections.

#### Productivity

In the training task, the average productivity of participants was 38.4 (SD=11.0) WPM. In the smartphone display conditions, this increased to 42.3 (SD=10.9) WPM, and 42.5 (SD=12.2) WPM in the glanceable display conditions. No significant difference was found in productivity between notification displays $(p = 0.978)$.

#### Interruptions

Figure 7.17 displays the resumption lag after ignoring or responding to a notification, with a summary of values in Table 7.1. As expected, reading a notification from the notification bar allowed the lowest response delay (4.39+/-2.97s). This was unsurprising, as the notification bar only required a single step to respond to a message after reading it. However, there was no significant difference between the notification bar, desktop and situated display when responding to a notification ($p > 0.1$). Additionally, the desktop (4.45+/-3.10s, $p < 0.001$) and situated display (4.80+/-3.84s, $p < 0.01$) were significantly faster to respond than the lockscreen (5.58+/-3.44s), and the smartwatch (5.38+/-3.88s, $p = 0.52$) was also faster, but this was not significant. This result is surprising, since these displays required more steps to respond to the notification than the lockscreen, where the device was already at hand. One possible explanation for this result is that participants were able to continue typing while reading the message before deciding to stop the typing task, and were able to quickly navigate to the notification bar without stopping to look at the lockscreen, yielding results similar to the notification bar. Overall, glanceable displays were not significantly slower than smartphone displays to respond to a notification ($p = 2.44$).

When ignoring a notification, the resumption lag was lowest with the desktop pop-up (1.86+/-1.48s), followed by the smartwatch (2.41+/-2.03s) and situated display (2.41+/-2.49s). The desktop pop-up was significantly faster than the situated display ($p < 0.001$) and the smartwatch ($p < 0.001$). No significant difference was found between the situated display and smartwatch ($p = 1.0$), or the lockscreen and smart eyewear ($p = 0.18$) when ignoring a notification. Overall, smartphone displays were significantly slower (5.03+/-2.75s) than glanceable displays (3.99+/-3.85s) when ignoring a notification ($p < 0.001$).

The resumption lag was largest (6.75+/-4.95s) after responding to a notification with the smart eyewear, followed by ignoring a notification with the notification bar (5.61+/-1.85s). This result implies that participants would be faster using the notification bar to respond or ignore notifications than using smart eyewear while typing. The desktop pop-up was significantly faster than all other notification displays when ignoring a notification, and was not significantly slower than the notification bar when responding to a notification. If the resumption lag of notifications that were ignored and responded are considered together, an average interruption lasted 3.14s with the desktop pop-up, and 5.00s with the notification bar. Therefore, the desktop pop-up was the most efficient for negotiating smartphone notifications overall, resulting in an average saving of 1.86s per notification compared to using the notification bar.

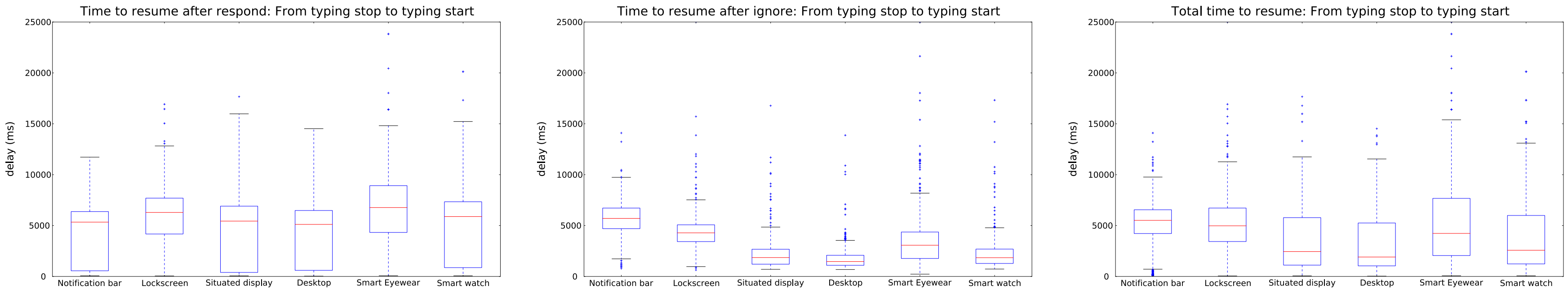
### Subjective Opinion

Subjective ratings are displayed in Figure 7.18. Participants rated smartphone displays significantly less convenient ($p < 0.001$), less easy to read content ($p < 0.001$) or concentrate ($p < 0.001$), and more frustrating ($p < 0.001$). The most common qualitative points that participants made are presented for each display in the following sections.

**Notification bar.** With the notification bar, many participants liked being "able to respond easily using the display method" [P6] and "it is a common and straightforward method for reading notifications" [P25], but disliked "having to unlock and manually look at every notification even though not all of them you needed to respond to. Also having to do that each time broke my rhythm of typing and made me lose my place a few times" [P3]. Many participants acknowledged that "at the moment for most people this all there is available" [P1], and "I do this all the time" [P15]. Participants also reported that they use the notification bar in "everyday life, but not because I would choose this method, but due to lack of better one" [P17] and "this method [is used] in my everyday life and I'm tired of it" [P20].

**Lockscreen.** With the lockscreen, participants "liked the fact that I could see the notification without unlocking the phone" [P10] and "it was faster than using the basic notification bar for non-urgent notifications" [P11], but disliked that "I still had to turn on my display to read it, distracting me from whatever I was doing" [P20] and "you still had to take your hands off the keyboard" [P27]. The lockscreen could be used "pretty much anywhere, even though it might be quite obvious you're checking your phone" [P22] and "when I need a glance overview of notifications (check periodically without unlocking phone)" [P8]. Some participants reported that "I do use this as my default, I suppose. But I don't exactly like it" [P9] and would use the lockscreen "only where no other option was available" [P6].

**Situated display.** Participants "liked that [the situated display] didn't interrupt what I was doing so much and felt it was easier to concentrate. I also felt it was less urgent to pick up the smartphone as I knew what the notification was about already" [P13]. Several participants reported that "you just need a glance at the screen to see whether you need to respond to the message" [P5], and "I could glance at the display to check the notification without my hands leaving the keyboard" [P14]. Participants felt that it was "much faster than the smartphone since I just had to slightly turn my head around" [P2] and "easy to manage to concentrate on writing at the same time" [P22]. However, participants disliked that "The fact that the background changed meant that I sometimes thought that something was trying to get my attention over there when it maybe wasn't" [P9] and "because the display was very big, it was hard to read the messages and therefore took some time to read it and was easily distracted" [P18]. Participants could imagine using the situated display "at home or at a workplace" [P24], "at the office or in my flat/at a party" [P3], and "I would definitely use it either when I am working at home or at the office since I receive notifications regularly" [P20].

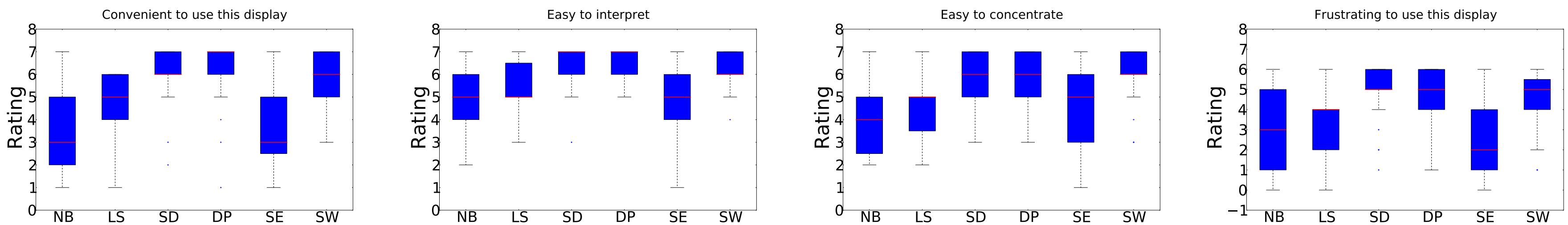



(a) Responding.

(b) Ignoring.

(c) Responding or ignoring.

Figure 7.17: Resumption lag when responding or ignoring a notification. It was faster to resume the typing task after ignoring a notification with the desktop pop-up, and slowest with the notification bar.




(a) Convenient to read the notification.

(b) Easy to read the notification.

(c) Easy to concentrate on the typing task.

(d) Frustrating to read the notification (axis inverted for clarity).

Figure 7.18: Ratings of subjective opinion. The situated display, desktop pop-up, and smartwatch are rated significantly higher than the notification bar, lockscreen and smart eyewear.

**Desktop pop-up.** Participants liked that the desktop pop-ups made it "very convenient to look at the notification because it's literally on the screen I'm already looking at" [P24], "was easy to casually glance and check" [P6], and "everything's all in the one place. You only need to look away from the screen if the message is urgent" [P7]. Some participants found that "it seems to me that it's a bit disturbing when you are working, as notifications will frequently pop up on the same screen" [P5], "Might be an issue if popup come in front of my work" [P16] and "it's not bad, however if I was trying to concentrate in real life it can be distracting" [P7]. P13 "felt that I had to respond more urgently than the prior experiment [situated display] as previously it was to the side of me and somehow I could almost prioritise more easily and finish my typing whereas when it was on the monitor I felt I had to do it right then". Desktop pop-ups could be used in a "daily working situation" [P5], while "working on a desktop PC in a workplace where others won't have the need to look at your screen" [P1], and "it is very useful for any time I using a computer. Though it is quite distracting while I am supposed to be working" [P14]. Some participants "use something similar already (PushBullet)" [P27] or were "just about to install AirDroid on my computer. I think that's what it does. Very very helpful thing to have!" [P15].

**Smart Eyewear.** Wearing the smart eyewear, participants liked "not having to move my hands from the keyboard" [P11], and "only I could see the notifications. Better privacy than when notifications are displayed on your laptop" [P15]. However, participants found that the "head gesture feels very unnatural and awkward, gesture detection was inconsistent leading me to view the notification on the phone instead" [P24]. Participants did not like "The narrow field of view" [P3], "having to move my head up to trigger the display'' [P11] and "the notification did not appear immediately on the Glass screen" [P26]. Participants could imagine using eyewear while "doing a task that involved more complicated use of your hands than typing, like if you were wearing gloves or something that would make it harder to check on your phone" [P11], "in case I am driving or have a very limited liberty of movement with my hands" [P2] and "when outside walking about, not necessarily when inside using a desktop computer" [P7] and "I would be nice when I was carrying out a task where I needed to pay attention to it, but where there wasn't already a screen in front of me (cooking, or walking, etc)" [P9].

**Smartwatch.** Participants liked that the smartwatch "was outside your vision so wasn't a distraction. The vibration was very short - let you know something was there but left you alone quickly" [P8], and "the notifications waited on you to be ready for them, so they didn't distract you" [P27]. P1 liked that "I tend to look at my hands while typing, so it was easy to glance down to my wrist (easier than other display methods). I also liked that the was a vibration I could feel when notifications came in" [P1]. Participants found the smartwatch to be "less distracting that I thought it would be. It was easy to glance at the screen and understand the notification without moving too much" [P9] and "with my hands

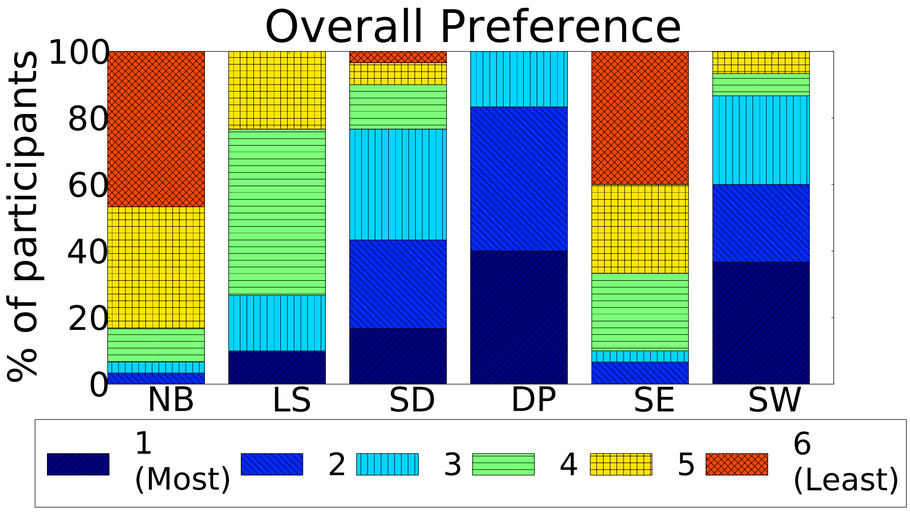



Figure 7.19: Overall preference. (Left-to-Right: Notification Bar *(NB)*, Lockscreen *(LS)*, Situated Display *(SD)*, Desktop Pop-ups *(DP)*, Smart Eyewear *(SE)*, Smartwatch *(SW)*). Desktop pop-ups are most preferred, and the notification bar is least preferred.

on the keyboard I didn't have to move my hands, just drop my eyes on the watch.. also that the watch lights up itself to show the notification" [P17]. Some participants felt that "it was annoying that you had to lift your hand to read the notification. This meant stopping typing. It was also quite heavy on the wrist, so it was a bit annoying to type with" [P14], ''it was a little frustrating typing in a keyboard while wearing a watch but that's probably because I am not used to it" [P10] and "actually having to move my arm when I am typing" [P12]. Participants could imagine using the smartwatch "while I am hanging out with friends I would prefer using a fast method to check my notifications like this, rather than having to take out the phone of my pocket" [P10], "receiving notifications in public when wanting to be discreet" [P1] and "if I was out and didn't want to constantly check my phone in my pocket" [P14].

### Overall Preference

The ranking of each display is displayed in Figure 7.19. The desktop pop-up was preferred most overall (12/30). In final comments, participants shared that "pop-ups were the best cause it was the quickest way to decide whether the notification was important or not" [P15], "the desktop pop-up required least effort to read" [P14] and "really liked the desktop and situated display as it was easy to interpret" [P30].

The desktop pop-up was not preferred significantly more than the smartwatch ($p = 0.083$), which was ranked second most preferred. Many participants (10/30) preferred the smartwatch most. In the final comments, P3 shared that "the watch was my favourite, with its bright notification screen it was handy for seeing what needed my attention and did not interrupt with the task", and P7 stated "need to get me a smart watch". The preference of smartwatch was surprising, as few participants expected to choose a wearable in the preliminary questionnaire. In addition to the concerns over the comfort of wearing a smartwatch while typing, visual appeal was also considered to be important. P15 felt that "the watch was great as well [but] the notifications were not that well displayed on it though". Additionally, the smartwatch was not significantly preferable to the situated display ($p = 0.056$). P9 shared that "I only choose the situated display over the desktop popup because the notifications were formatted more pleasantly", and P11 commented that "the auto changing background on the situated display was a little distracting, if I was using one I'd prefer to have a static background so it didn't catch your eye when it changed".

The notification bar was least preferred, despite the majority of participants choosing smartphone displays in the preliminary questionnaire. Interestingly, the notification bar was not preferred significantly less than smart eyewear ($p = 0.169$). Participants felt that "the smart eyewear required the most [effort to read]' [P14], 'eyewear was different to use but a bit distracting" [P30], and "apart from the Google Glass device, all notification methods seemed quite easy to read without disrupting my current task very often" [P26].

Some participants were frustrated that the response task required using the notification bar. P1 felt that "the ideal response would be to answer the message verbally by speaking into the device". P9 acknowledged that the choice of notification display depended on the context, and that "some of the notification displays would have ranked differently had I not been using a computer at the time. E.g. the watch and glass would have been better if I was doing a different task (cooking, walking...pottery, whatever) where I didn't have a screen directly in front of me. When the main screen is there, I think the extra displays are a bit unnecessary". Figure 5.13 highlights the application of a situated display as a way for multiple smartphone users to negotiate interruptions in a social context. Overall, participants felt that the "experiment was very enjoyable" [P30], "it was fun" [P19] and they were "pretty interesting ways of getting notifications" [P22].

### 7.3.5 Implications for Design

*Fewer actions to ignore is better than fewer actions to respond.* The notification bar required many interactions and quickly became irritating, especially when the notification could be ignored. In comparison, the desktop pop-up required only 2 steps to ignore a notification, and was preferred most overall. Additionally, the desktop pop-up was not significantly slower than the notification bar to respond, despite requiring 8 additional steps to respond to the notification.

*Glanceable notifications are most suitable when focusing on a task.* The desktop pop-up was most preferred as participants could read the notification with little movement and no interaction. The smartwatch and situated display were also valued for this feature. In particular, participants appreciated the notifications making use of existing displays that are required for the main task. However, the smart eyewear was not rated the same as the other three displays, as highlighted in the following point.

*Glanceable notification displays must be reliable to negotiate interruptions.* Participants disliked that Google Glass did not display the notification if they waited too long, and found the head-up feature to be unreliable. This greatly reduced the user experience of negotiating interruptions with this notification display. Future work should improve this comparison with a more reliable trigger, such as a peripheral display [34] or finger input [100].

*A subtle notification display is more socially acceptable than an intrusive one.* The smartwatch was considered to be a fast and discreet way of reading notifications, and was considered more appropriate in a public or social setting. Participants reported that they would prefer to use a smartwatch instead of their smartphone display in this context.

*Privacy in notification displays will depend on context.* Several participants noted privacy issues, including not wanting their smartphone notifications to appear when others are looking at the desktop PC. The context of the user, including who is near the display, will have an impact on the choice of a notification display. Future work should consider how a user might manage multiple notification displays that react to the context of the user.

### 7.3.6 Discussion

A comparison of six smartphone notification displays and their impact on attention to a typing task was presented. External displays were rated significantly higher than smartphone displays overall, despite requiring more actions to respond to a notification. The data collected allowed more detailed analysis of notification display choice on typing performance and a worst-case estimate of resumption lag.

Desktop pop-ups were found to be most efficient, saving 1.86s on average than the notification bar, and was preferred most overall. The smartwatch and situated display closely followed desktop pop-ups in preference and efficiency, and glanceable displays were more preferable than smartphone displays overall, despite requiring more steps to respond to a notification. The notification bar was least preferred, and reduced performance most when a notification could be ignored, but was the fastest way to respond. In comparison, smart eyewear was not significantly more preferable to the notification bar, and increased the average resumption lag by 0.36s when the number of response and ignore tasks were equal. Under these conditions, it would be better to choose the notification bar than eyewear as presented in this study. However, as more notifications are usually ignored than responded to, any display that does not involve the notification bar could increase the decrease the resumption lag overall.

Due to the uncertainty of which display the user was attending to during the typing task, the measure of resumption lag was a worst-case estimate. Despite the limitations of visual tracking to detect a visual attention switch to a notification display, the approach proved to be promising. With the release of more reliable eye tracking solutions, such as the Tobii Glasses[4], it will soon be possible to gather more reliable quantitative evidence to compare resumption lag with a set up similar to the experiment presented.

This work was a controlled pre-cursor to the application of notification displays in social scenarios. Future work should consider the impact of smartphone notification display choice on attention to social situations and in more natural settings with personal devices.

## 7.4 Summary

This chapter considered ways of designing interactions with a situated display that support social engagement and the negotiation of notifications in places (RQ-3 of Section 3.1).

Cast Together was designed as a probe to investigate inclusive and unobtrusive mobile interactions, by allowing connected smartphones to automatically co-ordinate music and photos on the display, and to share notifications and app launches so that they can be easily reviewed by co-located persons. The hedonic and pragmatic design of Cast Together was evaluated in a questionnaire. Three groups of 4 participants completed the questionnaire after experiencing Cast Together during a short board game session. The results show that Cast Together delivers an inclusive and unobtrusive experience.

A user study followed the experience of the author and two close friends who used Cast Together in their dynamic social situations: intimate home and office environments, social parties and board game events and during a family holiday. The feedback from these case studies provides initial insights to the use of a shared display in personal places.

A lab experiment with 30 participants evaluates Cast Together in a lab experiment by comparing notification displays while focusing on a typing task. Three groups of notification displays were compared: smartphone displays (notification bar, lockscreen), external displays (desktop, situated display) and wearables (smart eyewear, smartwatch). Cast Together was considered appropriate for in an office or a home environment, and confirmed its design as a social display.

[4]`http://www.tobii.com/`

# Chapter 8

# Conclusions

This chapter revisits the research aims and provides a summary of the contributions made by this thesis. A discussion of the results and implications for design of utilising presence in places to support mobile interaction around co-located social situations and primary tasks. The rapid prototyping approach taken has several limitations, and further work is required to generalise these contributions. The challenges that remain and the opportunities to build on this work are presented.

## 8.1 Research Questions

Four research questions were explored, with the aim to support mobile interaction in personal places:

1. How might mobile interaction be situated around artefacts of personal places, and support users to access content from their smartphone while managing their physical presence?
2. How might adapting menus to personal places reduce the time and effort of app navigation on the smartphone, and increase self-reflection on where apps are used?
3. How might coordinating smartphone content on a situated display support social engagement and the negotiation of notifications?
4. What are the capabilities and limitations of a rapid-prototyping approach with the Microsoft Kinect depth sensor and Bluetooth beacons to detect the smartphone in places?

## 8.2 Contributions

### 8.2.1 Adapting Menus to Mobile Information Needs in Places

A contribution was made to adaptive menu design on the mobile homescreen. In Chapter 6, an app launch and notification dataset was collected in the context of places, and provides quantitative evidence to suggest that app habits can relate to personal places. This dataset can be explored in an interactive visualisation, and is available to be further analysed by others. The app launch dataset enabled the evaluation of an adaptive homescreen menu in a lab experiment and a user study. Results from an experiment of adaptations on the mobile homescreen provides evidence that users prefer apps to be ordered by rank when movements are large, and was published in the International Journal of Human Computer Interaction [115]. This contribution provides a first step to automatically gathering app launch data in indoor places.

### 8.2.2 Inclusive and Unobtrusive Interaction with a Collaborative Media Display

A contribution was made to the pervasive display community. In Chapter 7, evaluations of a collaborative media sharing display in a social board game situation and primary typing task was described. Results from a social board game evaluation provide evidence that coordinating events from a smartphone automatically can reduce direct interaction with a private display, and sharing media can stimulate conversation and encourage an inclusive user experience. Results from an evaluation of notification displays in a typing situation provides qualitative insights from smartphone users who prefer a faster approach to reading a notification while managing a primary task, compared to a faster method of responding to a notification. The design of the Cast Together probe was published as a poster at the Pervasive Display conference [113], and the result of the typing scenario was published as a workshop paper at the Mobile Human Computer Interaction conference [114], and was extended for the International Journal of Mobile Human Computer Interaction [116].

### 8.2.3 Utilising Presence in Places

Four place-aware prototypes were described in Chapters 5.5, 6 and 7 that demonstrate interactions between: web pages and physical objects, digital books and physical structures, smartphone apps and physical places, and digital media and co-located people. Each prototype enabled user evaluations to be performed with the context of physical places, and are available to be built on by others. Results from user evaluations highlight opportunities to increase the visibility of private smartphone interactions by interacting around objects and structures, and overcome the discoverability of invisible interfaces by utilising personal relationships with places. Movement around personal places was demonstrated to reduce visual attention to a private display. This design of movement to interact with a digital bookshelf application was published as a poster at the Mobile and Ubiquitous Multimedia conference [110].

### 8.2.4 A Rapid Prototyping Approach to Sensing Places

Two rapid prototyping tools were described in Chapter 4 to integrate the sensing of the user environment with a smartphone. Tools were built with the low-cost sensors that are currently available: the Microsoft Kinect enables the tracking of fine-grained movements in a small area, and Low-Energy Bluetooth beacons can be detected by the smartphone and assigned to logical places. Each tool is designed with minimal set-up costs to enable rapid exploration in personal environments. Results of positioning experiments with each tool highlight the capabilities and limitations of current sensing techniques for interacting with a smartphone in personal places. In addition to positioning and visual tracking, NFC was explored to tag physical objects in places with the Cast Together probe. The rapid prototyping approach to ubiquitous interaction with a mobile device and Kinect was published as a short paper at the Mobile Human Computer Interaction conference [112].

## 8.3 Future Work

### 8.3.1 Recommending Media to Co-Located People

There are opportunities to perform collaborative filtering based on the people who share a place and their personal preferences. For example, Netflix, the video streaming service, filters its vast collection of videos to recommend ones that it predicts a user will like but might not have seen. Cast Together demonstrates that multiple personalities can be projected on a situated display by combining public historical activity from social media profiles. A simple round-robin schedule was chosen to manage the songs and photos of connected users. By combining the profiles of other users who share a display, the collective preferences could help to find films that the group are most likely to enjoy, but might not have seen. Alternatively, a service could automatically find music that all nearby users have in common, and would enjoy listening to. Furthermore, if one person were to leave the group, a system could remove their preferences, and dynamically adapt to the preferences of users in the place. Collaborative filtering with co-located user profiles could improve user experience, while allowing users to disengage from their devices.

Privacy concerns are an inherent challenge to sharing personal information. An assumption was made that personal places are shared by people whom the user knows, which separates this work from public displays and location-aware services, as strangers might also share or visit a public place. However, real-life contexts can be complicated, and the privacy issues associated with making personal information public is still a challenge that needs to be faced before sharing smartphone events can generalise to every day life.

### 8.3.2 Utilising Presence in Places

There are opportunities to explore interactions with mobile applications that bridge the gap between the digital and physical by using movement around physical artefacts in a place. Evaluations in the wild will be required to understand the implications of making interactions visible, and the extent to which users will choose movement as a supplimentary form of interaction. The Virtual Sensors platform demonstrates that it is possible use the Kinect to sense interactions independent of the mobile device. Multiple device interactions are possible by using a single gesture or entering a certain region of a place. Similarly, multiple persons can interact with a single device by detecting their movement in a room. Future work should consider the opportunities of presence in personal place to help users manage content from their personal devices.

An insight from user evaluations with the Microsoft Kinect was that movement would only be suitable as a supplementary form of mobile interaction, and participants were unsure of when movement might be preferred over a touchscreen. Long-term evaluations will be required to understand the role of movement as an interaction technique for mobile users in personal places.

### 8.3.3 Advanced Sensing of Personal Environments

This research was motivated by the challenge of detecting the smartphone user in the physical environment. A rapid prototyping approach made it possible to leverage the low-cost sensing techniques currently available, and tools were developed to combine this sensing with a mobile device. Though the Microsoft Kinect provides low-cost motion tracking, it was found to be too unreliable for a robust system. Similarly, Low Energy Bluetooth beacons alone are not enough to reliably detect places indoors. The limitations of each sensing approach were accepted in order to explore the interactions that are possible when more advanced sensing techniques are available, and provides motivation to improve the development of low-cost positioning systems that make it easy for interaction designers to design interactions with the context of the smartphone user in personal places. Prototypes developed with a rapid prototyping approach will require more robust sensing to be used in the wild.

More advanced sensing techniques will be required to improve the accuracy of place detection. User studies will also be required to improve the usability of evaluation tools, and reduce the set up costs for users. Marking places with Bluetooth beacons could be used to learn place labels from other sensors. This could determine to what extent logical labels represent complex features, such as GPS location, time of day and previous app launches. Beacons could be used to train contextual features, which could simplify the deployment of place detection by relying on a small number of beacons for short-term while learning from a larger number of features.

## 8.4 Outlook

Sensing technologies are becoming as pervasive as mobile devices, and it will soon be possible to perform reliable place detection with smartphones. This thesis highlighted the opportunity to utilise presence in personal places to support mobile interaction. Adapting the mobile interface to personal places can reduce the time and effort of app navigation. Situating the mobile interface into the physical environment can create opportunities for inclusive interactions with a private display. Coordinating media and events from a smartphone to a situated display can support the negotiation of interruptions, and stimulate conversation without active engagement with a mobile device. Challenges still exist before presence in personal places can be fully utilised. Developers and interaction designers will require robust tools and frameworks to design better user experiences with mobile applications. The rapid prototyping approach taken in this thesis allows user experiences to be explored early in the design stage. Novel interactions can encourage users to manage their physical presence around the people and places that matter most in everyday life.

# Appendix A

# Questionnaire: Relationships Between Websites and Personal Places

# Relating Websites to Physical Places

We ask you to think of 10 websites that you use frequently. If you are unsure, you can check your personal web history on your browser, your bookmarks folder or look for 'most visited' on Chrome.

We will then ask you to think about something in your physical places that you think best represents each website.

There are a total of 15 questions and the form should take no longer than 15 minutes to fill out.

*Required

1. **Specify the name of 10 websites that you visit often and an artefact for each that you would represent it with in your physical environment.** *
   i.e. website - artefact

2. **Do you currently own a smartphone?** *

3. **How frequently do you use your smartphone to browse the Internet?** *

4. **How frequently do you use any device to browse the web?** *

5. **Age range.**

6. **Gender.**

# Appendix B

# Questionnaire: Object Tagging with a Mobile Web Browser

# Spatial Web Browser Experiment

*Required

1. **Dominant hand** *

## Scenario 1

Please answer the questions in the order requested by the experimenter.

2. **To which artefact did you assign the url [www.amazon.co.uk/Romance-Books/b?node=88](www.amazon.co.uk/Romance-Books/b?node=88)** *

3. **How did you find where you had placed the website in the space?** *

## Scenario 2

4. **To which artefact did you assign the url [www.vimeo.com/channels/robots](www.vimeo.com/channels/robots)?** *

5. **How did you find where you had placed the website in the space?** *

## Experience

Rate the following from 1=strongly disagree to 7=strongly agree.

6. **I was confident about where I placed the webpages in the space.** *

7. **I would remember which artefact I attached a webpage to over time.** *

8. **The artefacts in the space provided me with plenty of options to store the webpages.** *

9. **I can imagine attaching web pages to artefacts in my environment.** *

10. **I can imagine retrieving web pages on my mobile device from artefacts in my environment.** *

11. **Additional comments** *

## Demographic information

12. **Gender.**

13. **Age bracket.**

Powered by
Google Forms

# Appendix C

# Questionnaire: App Tracking in Places

# AppWhere Mid-point Questionnaire

Please answer a series of questions to help us understand your mobile phone use. There are 20 questions and the form should take no longer than 15 minutes to complete.

*Required

## Smartphone experience

1. **How long have you been using your current smart phone? ***

2. **How long have you been using an Android smart phone? ***

3. **How experienced are you with your smartphone? ***
   (1=novice, 5=expert)

4. **Do you use apps to customise your phone? If so, why? ***

## Your apps

5. **How many apps do you use daily? ***

6. **What category of apps do you use daily? ***

7. **How many apps have you installed in the past 2 weeks? ***

8. **How often do you install apps?**

9. **How many apps have you uninstalled in the past 2 weeks?** *

10. **How often do you uninstall apps?** *

## Your home screen

The homescreen is a collection of pages that display a selection of shortcuts to your apps.

11. **For which reasons do you arrange icons on your homescreen?** *

12. **How are icons on your homescreen arranged?** *

13. **How often do you arrange icons on your homescreen?** *

14. **What do you think about your homescreen?** *

15. **I remember which apps are on my home screen** *
(1=strongly disagree, 5 =strongly agree)

16. **I always check my home screen when I want to launch an app** *
(1=strongly disagree, 5 =strongly agree)

17. **I always check the front page of the home screen when I want to launch an app** *
(1=strongly disagree, 5 =strongly agree)

18. **I do not use my home screen** *
(1=strongly disagree, 5 =strongly agree)

19. **I always check the full app list when I want to launch an app, instead of the homescreen** *
(1=strongly disagree, 5 =strongly agree)

## Your places

Please answer a few questions about where you use your phone.

20. **In which places do you use your phone most?** *

21. **Do you use different apps in your personal places compared to in public? Why?** *

## Demographic Information

22. **Gender**

23. **Age range**

# Appwhere - Final

Please answer a few questions about the adaptive widget that was placed on your home screen in the third week of the experiment.

*Required

1. **How aware of the adaptive changes to the widget were you?** *
   *Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Unaware | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Very aware |

2. **How useful were the adaptive changes to the widget?** *
   *Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Not useful | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Very useful |

3. **How predictable were the adaptive changes to the widget?** *
   *Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Random | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Predictable |

4. **How satisfied were you with the adaptive changes to the widget?** *
   *Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Not satisfied | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Very satisfied |

5. **The adaptive changes were useful in helping me complete the tasks.** *
   *Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

6. **I felt in control of the adaptive changes to the widget.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

7. **The adaptive changes made it frustrating to use the widget.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

8. **The adaptive changes made me efficient in completing the tasks.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

9. **I used the adaptive widget to launch app shortcuts often.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

10. **I would use an adaptive widget for my app shortcuts in the future.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

11. **Please provide any additional feedback below.**

..........................................................................................

# Appendix D

# Questionnaire: Stability in an Adaptive Homescreen

# Adaptive Homescreen Experiment - Preliminary
*Required

1. **Please rate your experience with smartphones** *
   (1=no experience, 7=very experienced)

2. **Please rate your experience with Android smart phones** *
   (1=no experience, 7=very experienced)

## Demographic Information

Please tell us a little about yourself so that we can better understand our results.

3. **Gender** *

4. **Age Range** *

# Adaptive Homescreen Experiment

*Required

1. **How aware of the adaptive changes to the widget were you?** *
   (1=very unaware, 7=very aware)

2. **How useful were the adaptive changes to the widget?** *
   (1=not useful, 7=very useful)

3. **How predictable were the adaptive changes to the widget?** *
   (1=random, 7=very predictable)

4. **How satisfied were you with the adaptive changes to the widget?** *
   (1=not satisfied, 7=very satisfied)

5. **The adaptive changes were useful in helping me complete the tasks** *
   (1=strongly disagree, 7=strongly agree)

6. **I felt in control of the adaptive changes to the widget** *
   (1=strongly disagree, 7=strongly agree)

7. **The adaptive changes made it frustrating to use the widget** *
   (1=strongly disagree, 7=strongly agree)

8. **The adaptive changes made me efficient in completing the tasks** *
   (1=strongly disagree, 7=strongly agree)

9. **Additional comments**

Powered by
Google Forms

# Adaptive Homescreen Experiment - Final

*Required

1. **Preference 1** *

   Most preferred

   *Mark only one oval.*

   - Most Frequently Used Next - Ordered alphabetically
   - Most Frequently Used Next - Ordered by most frequent
   - Most Recently Used - Ordered alphabetically
   - Most Recently Used - Ordered by time

2. **Preference 2** *

   *Mark only one oval.*

   - Most Frequently Used Next - Ordered alphabetically
   - Most Frequently Used Next - Ordered by most frequent
   - Most Recently Used - Ordered alphabetically
   - Most Recently Used - Ordered by time

3. **Preference 3** *

   *Mark only one oval.*

   - Most Frequently Used Next - Ordered alphabetically
   - Most Frequently Used Next - Ordered by most frequent
   - Most Recently Used - Ordered alphabetically
   - Most Recently Used - Ordered by time

4. **Preference 4** *

   *Mark only one oval.*

   - Most Frequently Used Next - Ordered alphabetically
   - Most Frequently Used Next - Ordered by most frequent
   - Most Recently Used - Ordered alphabetically
   - Most Recently Used - Ordered by time

5. **Additional comments**

   ..................................................

---

# Appendix E

# Questionnaire: Appwhere in Places

# Appwhere - End of Experiment

Please answer a few questions about the Appwhere experiment. This questionnaire will have 3 pages on the topics of the My Places and Appwhere apps, and the adaptive homescreen widget. It should take no longer than 30 mins to complete.

*Required

1. **Gender**

2. **Age**

3. **How long have you been using an Android smartphone?**

## Places

This section is about the My Places app and the Bluetooth beacons that you were provided with to mark each of your personal places.

## Bluetooth Beacons

The Kontact.io Bluetooth beacons were small white boxes that you were asked to put in each of your places.

4. **What did you think about using the Bluetooth beacons to identify your places?** *

5. **If you put Bluetooth beacons in your places, in which places did you find to be most useful?** *

6. **Were there any places that you would have liked to have put Bluetooth beacons in?** *

7. **I did not like having Bluetooth beacons in my places** *
   (1=strongly disagree, 5 =strongly agree)

8. **I forgot about the Bluetooth beacons in my places.** *
   (1=strongly disagree, 5 =strongly agree)

## My Places app

The My Places app allowed you to assign a name to each Bluetooth beacon. These names appeared in the Appwhere app and adaptive widget.

9. **I added Bluetooth beacons with the My Places app** *

10. **What did you think about using the My Places app to label the Bluetooth beacons in your places?** *

11. **What did you think about the accuracy of the My Places app?** *

12. **What did you think about the My Places notification that appeared when you entered a place?** *

13. **The My Places app was accurate in detecting which of my places I was in** *
    (1=strongly disagree, 5 =strongly agree)

14. **I was aware of the My Places notification** *
    (1=strongly disagree, 5 =strongly agree)

15. **I found the My Places notification to be useful** *
    (1=strongly disagree, 5 =strongly agree)

## Overall

16. **I am more aware of where I use my device after using My Places.** *
(1=strongly disagree, 5 =strongly agree)

17. **I felt self-conscious about using My Places.** *
(1=strongly disagree, 5 =strongly agree)

18. **I would use Bluetooth beacons to mark my places in the future.** *
(1=strongly disagree, 5 =strongly agree)

19. **I would use an app like My Places to detect my places in the future.** *
(1=strongly disagree, 5 =strongly agree)

20. **Do you have any other comments about the Bluetooth beacons or My Places app?**

## Appwhere

This section is about the Appwhere app and homescreen widget that tracked your app launches and displayed statistics on how often you used apps in your places.

## App Tracking

The app tracking feature logged your app launches in the background.

21. **What did you think about the app tracking feature of the Appwhere app?** *

22. **I forgot about the app tracking feature.** *
(1=strongly disagree, 5 =strongly agree)

23. **I did not like the app tracking feature.** *
(1=strongly disagree, 5 =strongly agree)

## Statistics

The statistics in the Appwhere app displayed a summary of your app use in each of your places over the past week.

24. **I checked my app use statistics with Appwhere.** *

25. **What did you think about the statistics in the Appwhere app?** *

26. **If you checked the Appwhere statistics, were there any occasions in particular that you checked them?** *

27. **Did you learn anything new about your app use through the Appwhere statistics?** *

28. **I was aware of the statistics in the Appwhere app.** *
    (1=strongly disagree, 5 =strongly agree)

29. **I was satisfied with the statistics in the Appwhere app.** *
    (1=strongly disagree, 5 =strongly agree)

30. **I found the statistics in the Appwhere app to be useful.** *
    (1=strongly disagree, 5 =strongly agree)

31. **Were there any statistics or features that you felt were missing in the Appwhere app?** *

32. **Can you think of any way that Appwhere statistics changed the way that you used your mobile device?** *

## Photo

The photo feature allowed you to set a photo to represent each of your places. This photo was displayed in the homescreen widget.

33. **I set a photo for my places with the Appwhere app.** *

34. **What did you think about the photo feature of the Appwhere app?** *

35. **If you set a photo, how did you choose one?** *

36. **I was aware of the photo feature.** *
(1=strongly disagree, 5 =strongly agree)

37. **I was satisfied with the photo feature** *
(1=strongly disagree, 5 =strongly agree)

38. **I found the photo feature to be useful** *
(1=strongly disagree, 5 =strongly agree)

## Excluding apps

The exclusion feature allowed you to blacklist apps from appearing in the Appwhere statistics and homescreen widget.

39. **I excluded apps in the Appwhere app** *

40. **What did you think about the exclusion feature of the Appwhere app?** *

41. **If you excluded apps in Appwhere, for what reasons did you want to exclude them?** *

42. **I was aware of the exclusion feature.** *
(1=strongly disagree, 5 =strongly agree)

43. **I was satisfied with the exclusion feature.** *
(1=strongly disagree, 5 =strongly agree)

44. **I found the exclusion feature to be useful.** *
(1=strongly disagree, 5 =strongly agree)

## Overall

45. **I am more aware of which apps I use on my device after using Appwhere.** *
(1=strongly disagree, 5 =strongly agree)

46. **I am more aware of how often I use my device after using Appwhere.** *
(1=strongly disagree, 5 =strongly agree)

47. **I felt self-conscious about using Appwhere.** *
(1=strongly disagree, 5 =strongly agree)

48. **I would use an app tracker like Appwhere to log my app launches in the future.** *
(1=strongly disagree, 5 =strongly agree)

49. **Do you have any other comments about the Appwhere app?**

## Adaptive Homescreen

This section is about the homescreen widget that adapted your app icons to your places.

## Homescreen Widget

The homescreen widget arranged your app icons automatically

50. **I placed the widget on my homescreen** *

51. **What did you think about the homescreen widget?** *

52. **If you used the homescreen widget, when did you find it to be most useful?** *

53. **I did not like the adaptive homescreen widget.** *
(1=strongly disagree, 7=strongly agree)

54. **I knew when an app would be in the homescreen widget.** *
(1=strongly disagree, 7=strongly agree)

55. **I found the adaptive homescreen widget to be useful.** *
(1=strongly disagree, 7=strongly agree)

56. **If you also used the My Places app, how did you feel about the homescreen widget reacting to your places?** *

57. **If you also used the My Places app, were there any apps that you knew would be in the widget?** *

58. **Can you think of any way that the adaptive homescreen widget changed the way that you used your mobile device?** *

## Adaptive changes

The homescreen widget adapted your app icons automatically to show you the icons that you are most likely to launch next.

59. **What did you think about the apps updating in the widget?** *
E.g. changes to the order of apps and the selection of apps.

60. **If you noticed apps updating in the adaptive widget, were there any occasions in particular that you noticed them?** *

61. **I was aware of the adaptive changes to the widget.** *
(1=strongly disagree, 7=strongly agree)

62. **I found the adaptive changes to be frustrating** *
(1=strongly disagree, 7=strongly agree)

63. **The adaptive changes to the widget were useful.** *
(1=strongly disagree, 7=strongly agree)

64. **I found the adaptive changes to the widget to be predictable.** *
(1=strongly disagree, 7=strongly agree)

65. **I felt in control of the adaptive changes to the widget.** *
(1=strongly disagree, 7=strongly agree)

## Overall

66. **I could access my apps more quickly when I used the adaptive widget.** *
(1=strongly disagree, 7=strongly agree)

67. **I felt self-conscious about using the adaptive widget.** *
(1=strongly disagree, 7=strongly agree)

68. **I would use an adaptive widget to organise my app shortcuts in the future.** *
(1=strongly disagree, 7=strongly agree)

69. **Do you have any other comments about the adaptive homescreen?**

## Final comments

70. **Do you have any other comments about this experiment?** *

# Appendix F

# Questionnaire: User Experience of Cast Together

# Cast Together: User Experience Questionnaire

Please complete the following questions about using Cast Together, the shared media display on the television, while in a social situation, playing a board game with friends.

There are 4 pages with 7 bi-polar questions in each page. At the end you will be provided with a text box to enter any comments that you may have.

This questionnaire should take no longer than 10 minutes to complete.

*Required

## PQ

Please rate the following word pairs.

1. **Complicated or Simple** *
   *Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Complicated | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Simple |

2. **Technical or Human** *
   *Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Technical | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Human |

3. **Cumbersome or Straightforward** *
   *Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Cumbersome | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Straightforward |

4. **Impractical or Practical** *
   *Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Impractical | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Practical |

5. **Unruly or Manageable** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Unruly | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Manageable |

6. **Unpredictable or Predictable** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Unpredictable | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Predictable |

7. **Confusing or Clearly structured** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Confusing | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Clearly structured |

## HQ-I

Please rate the following word pairs.

8. **Unprofessional or Professional** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Unprofessional | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Professional |

9. **Isolating or Connective** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Isolating | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Connective |

10. **Cheap or Premium** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Cheap | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Premium |

11. **Tacky or Stylish** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Tacky | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Stylish |

12. **Unpresentable or Presentable** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Unpresentable | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Presentable |

13. **Alienating or Integrating** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Alienating | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Integrating |

14. **Separates me or Brings me closer** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Separates me | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Brings me closer |

## HQ-S

Please rate the following word pairs.

15. **Unimaginative or Creative** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Unimaginative | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Creative |

16. **Conventional or Inventive** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Conventional | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Inventive |

17. **Conservative or Innovative** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Conservative | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Innovative |

18. **Cautious or Bold** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Cautious | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Bold |

19. **Ordinary or Novel** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Ordinary | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Novel |

20. **Dull or Captivating** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Dull | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Captivating |

21. **Undemanding or Challenging** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Undemanding | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Challenging |

## ATT

Please rate the following word pairs.

22. **Bad or Good** *
*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Bad | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Good |

23. **What did you not like about using Cast Together?** *

________________________________

24. **Do you have any comments about using Cast Together?** *

25. **Discouraging or Motivating** *

*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Discouraging | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Motivating |

26. **Rejecting or Inviting** *

*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Rejecting | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Inviting |

27. **Repelling or Appealing** *

*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Repelling | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Appealing |

28. **Disagreeable or Likeable** *

*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Disagreeable | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Likeable |

29. **Ugly or Attractive** *

*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Ugly | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Attractive |

30. **What did you like about using Cast Together?** *

31. **Unpleasant or Pleasant** *

*Mark only one oval.*

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | |
|---|---|---|---|---|---|---|---|---|
| Unpleasant | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Pleasant |

# Appendix G

# Questionnaire: Cast Together in Places

# Situated Display Experiment - Preliminary

Prior to starting the Situated Display experiment, we would like to understand your current experience with using your mobile device in your office. This questionnaire is split into 5 pages with 7 questions per page on the following topics: General, Apps, Notifications, Music and Photos. It should take 15 minutes to complete. We would be very grateful if you could fill it out!

*Required

## Demographic Information

1. **Gender**

2. **Age**

3. **How long have you been using an Android smartphone?** *

4. **On average, how many days per week do you go to your office?** *

## Social

This section aims to understand the social dynamics of your office environment.

5. **On average, how many other people do you share your office with?** *

6. **If there are other people in your office, how do you usually interact?** *

7. **On average, how many occasions per day do you speak to others in your office?** *

8. **I work with the people in my office directly.** *

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | | | | | | | | Strongly agree |

9. **I only speak to people in my office about work.** *

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | | | | | | | | Strongly agree |

# Apps

## Launching Apps

This section is about how you launch apps in your office and the purposes that you use your mobile phone at work.

10. **Which apps do you use most in your office?** *

11. **On average, how often do you use apps in your office per day?** *

12. **I only use apps in my office that are important to my work.** *

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | | | | | | | | Strongly agree |

## Social

This section is about how your app use relates to other people in your office.

Please consider this section carefully as it will help us to identify any issues that you may have with sharing the name of app launches on the situated display.

13. **It is useful for me to know what other people in my office are doing on their mobile device.** *

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | | | | | | | | Strongly agree |

14. **I would not mind if people in my office were to see the name of the app I am using.** *

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | | | | | | | | Strongly agree |

15. **I would not mind if people in my office were to see that I am using my device.** *

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | | | | | | | | Strongly agree |

# Notifications

## Receiving Notifications

This section is about when you receive an alert or notification on your mobile phone.

16. **Which apps do you receive notifications for most in your office?** *

.......................................................................................

17. **All notifications I receive in my office are related to my work.** *

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | | | | | | | | Strongly agree |

## Checking Notifications

This section is about when you are expecting to receive a notification.

18. **When you check your mobile device for notifications in your office, which apps do you look for?** *

.........................................................................................

19. **How often do you check your phone for notifications in your office per day?** *

.........................................................................................

20. **I always check my device when I receive a notification in my office.** *

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

21. **When you notice a new notification and you are in your office, how quickly do you check what it is?** *

*Mark only one oval.*

◯ I usually check it immediately.

◯ I usually finish my current task first before checking any notifications.

◯ I usually wait until I have finished all my tasks for the day before checking any notifications.

◯ Other: .........................................................................................

## Social

This section is about how your notifications relate to the people in your office.

Please consider this section carefully as it will help us to identify any issues that you may have with sharing notifications on the situated display.

22. **All notifications I receive in my office are relevant to people I work with.** *

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

23. **I would not mind if people in my office could see the notifications I receive.** *

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

24. **I would not mind if people in my office were to see the name of the app for the notifications I receive.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

# Music

## Listening to Music

This section is about the equipment and services that you use to listen to music in your office.

25. **If you listen to music in your office, how do you play it?** *

..................................................................................

26. **I only listen to music in my office with headphones on.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

27. **Do you have a Last.fm account?** *
Last.fm is a free service that is popular for recording music listening history online.

..................................................................................

## Choosing Music

This section is about how you decide on the music to play in your office.

28. **If you listen to music in your office, how do you select it?** *

..................................................................................

29. **On average, how many times per day do you choose music to listen to in your office?**

..................................................................................

## Social

This section is about how the music in your office relates to the people around you. This will help us to identify any issues that you may have with playing music on the situated display.

30. **I would not want people in my office to know what I have been listening to recently.** *

*Mark only only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

31. **I know the music preferences of the people in my office.** *

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

32. **I would not want to listen to music selected by people in my office.** *

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

# Photos

## Sharing Photos

This section is about how you show photos to other people when you are in your office.

33. **If you share photos in your office, how do you usually share them?** *

34. **How often do you share photos in your office per day?** *

35. **Do you have a Flickr account?** *

Flickr is a free service that is popular for sharing photos online.

## Viewing Photos

This section is about how you look at photos in your office.

36. **If you look at photos in your office, how do you usually view them?** *
E.g. Facebook, email... Mobile device, desktop,...

37. **How often do you look at photos in your office per day?** *

## Social

This section is about how the photos that you share or look at in your office relate to the people around you. This will help us to identify any issues that you may have with sharing photos on the situated display.

38. **All photos I share in my office are with people I work with.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

39. **I would not want people in my office to see the photos that I have shared recently.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

40. **I would not want to look at photos shared by people in my office.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

# Week 1 of 4 - Situated Display Experiment

After the first week of the experiment, we would like to understand your experience so far with using the Cast Together display in your your office. The following questionnaire should take less than 30 minutes to complete. There are 5 pages with 10-15 questions per page. Questions are on the following topics: General, Apps, Notifications, Music and Photos.

*Required

1. **How many days this week did you go to your office?** *

   ..............................................................................

2. **On average, how many people did you share your office with this week?** *

   ..............................................................................

3. **Where did you use the situated display this week?** *

   ..............................................................................

4. **How did you use the situated display this week?** *

   ..............................................................................

5. **Did the situated display affect your interactions with others in your office?** *

   ..............................................................................

# Week 2 - Situated Display Experiment

We would like to understand your current experience with using your mobile device in your office after using the auto-connect feature this week. The following questionnaire should take less than 30 minutes to complete. There are 5 pages with 10-15 questions per page. Questions are on the following topics: General, Apps, Notifications, Music and Photos.

*Required

1. **How many days this week did you go to your office?** *

   ____________________

2. **On average, how many people did you share your office with this week?** *

   ____________________

3. **Where did you use the situated display this week?** *

   ____________________

4. **How did you use the situated display this week?** *

   ____________________

5. **Did the situated display affect your interactions with others in your office?** *

   ____________________

## Auto-connect

6. **Did you notice your device auto-connecting to Cast Together this week?** *

   ____________________

7. **I noticed my device automatically connect to the situated display every time I entered the room.** *

   *Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

8. **It was convenient when my device auto-connected to the situated display.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | | | | | | | | Strongly agree |

9. **I did not want my device to auto-connect to the situated display.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | | | | | | | | Strongly agree |

10. **What did you think about the auto-connect feature that came into effect this week?** *

# Week 3 - Situated Display Experiment

Prior to starting the Situated Display experiment, we would like to understand your experience with using the NFC tag feature this week. The following questionnaire should take less than 30 minutes to complete. There are 5 pages with 10-15 questions per page. Questions are on the following topics: General, Apps, Notifications, Music and Photos.

*Required

## General Information

1. **How many days this week did you go to your office? ***

2. **On average, how many people did you share your office with this week? ***

3. **Where did you use the situated display this week? ***

4. **How did you use the situated display this week? ***

5. **Did the situated display affect your interactions with others in your office? ***

## NFC tags

6. **What did you think about the NFC feature that came into effect this week? ***

7. **How did you use the NFC tags? ***
   E.g. Where did you place them? When did you scan them?

8. **My device updated its settings when I scanned an NFC tag.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

9. **I was aware of my device changing its settings when I scanned an NFC tag.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

10. **It was convenient to change my device settings by scanning an NFC tag.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

11. **I did not want to change my settings using an NFC tag.** *
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

# Week 4 - Situated Display Experiment

Prior to starting the Situated Display experiment, we would like to understand your current experience with using your mobile device in your office. The following questionnaire should take less than 30 minutes to complete. There are 6 pages with 10-15 questions per page. Questions are on the following topics: General, Apps, Notifications, Music, Photos and Final comments.

*Required

## General

1. **How many days this week did you go to your office? ***

2. **On average, how many people did you share your office with this week? ***

3. **Where did you use the situated display this week? ***

4. **How did you use the situated display this week? ***

5. **Did the situated display affect your interactions with others in your office? ***

## NFC tags

6. **What did you think about the NFC feature this week? ***

7. **I changed my Cast Together settings this week ***

8. **I used the NFC feature to change my Cast Together settings this week** *
*Mark only one oval.*

- ( ) Yes
- ( ) No

9. **I did not want to change my Cast Together settings using an NFC tag.** *
How strongly do you agree with this statement?
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ( ) | ( ) | ( ) | ( ) | ( ) | ( ) | ( ) | Strongly agree |

10. **If you used the NFC tags this week, what did you use them for?**

..........................................................................................

11. **My device updated its settings when I scanned an NFC tag.**
If you used the NFC feature this week, how strongly do you agree with this statement?
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ( ) | ( ) | ( ) | ( ) | ( ) | ( ) | ( ) | Strongly agree |

12. **I was aware of my Cast Together settings changing when I scanned an NFC tag.**
If you used the NFC feature this week, how strongly do you agree with this statement?
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ( ) | ( ) | ( ) | ( ) | ( ) | ( ) | ( ) | Strongly agree |

13. **It was convenient to change the Cast Together settings by scanning an NFC tag.**
If you used the NFC feature this week, how strongly do you agree with this statement?
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ( ) | ( ) | ( ) | ( ) | ( ) | ( ) | ( ) | Strongly agree |

14. **I would use NFC tags to change my device settings in the future.** *
How strongly do you agree with this statement?
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

## Bluetooth beacons

15. **What did you think about the autoconnect feature this week?** *

16. **I used the autoconnect feature this week** *

17. **I did not want my device to autoconnect to the situated display.** *
How strongly do you agree with this statement?
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

18. **My device always autoconnected to the situated display when I entered the room.**
If you used the autoconnect feature this week, how strongly do you agree with this statement?
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

19. **I was aware of my device autoconnecting to the situated display.**
If you used the autoconnect feature this week, how strongly do you agree with this statement?
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | ⬭ | Strongly agree |

20. **It was convenient when my device autoconnected to the situated display.**

If you used the autoconnect feature this week, how strongly do you agree with this statement?

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

21. **I would use an autoconnect feature like this to connect to a situated display in the future.** *

How strongly do you agree with this statement?

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

## Final comments

22. **The Cast Together app allowed me to use my mobile device less often.** *

How strongly do you agree with this statement?

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

23. **I would like to use an app similar to Cast Together in future.** *

How strongly do you agree with this statement?

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

24. **Do you have any final comments about this experiment?**

________________________________________

# Cast Together: Experience in Places

Please complete the following questions about a situation that you used Cast Together. You will be allowed to submit as many responses to this questionnaire for each situation that you used Cast Together.
There are 3 questions in total, and they should take no longer than 10 minutes to complete.
Please read all questions before completing your response.

*Required

1. **Participant id** *

2. **Which place will you be referring to?** *

3. **Which event, situation or time of day will you be referring to?** *

## User Experience

4. **Are there any scenarios that you remember Cast Together to have been helpful in this situation?** *

5. **Are there any scenarios that you felt Cast Together was unhelpful, or it did not meet your expectations in this situation?** *

6. **Do you have any further comments about using Cast Together in this situation?** *

# Appendix H

# Questionnaire: Choice in Notification Displays

# Notifications Experiment: Preliminary

Please complete the following questions. There are 6 questions in total and they should take no longer than 2 minutes to complete.

*Required

## Demographic Information

1. **Gender**

 ........................................................................

2. **Age**

 ........................................................................

## Notifications

3. **Which of the following do you have experience in reading your smartphone notifications?** *
 *Tick all that apply.*

 - [ ] Smart eyewear notifications (e.g. Google Glass)
 - [ ] Smart watch notifications (e.g. Android wear, pebble)
 - [ ] Android lockscreen notifications
 - [ ] Pop-up notifications on a Desktop PC or laptop
 - [ ] Notifications on a situated display
 - [ ] Android notification bar

4. **I receive smartphone notifications regularly.** *
 *Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ( ) | ( ) | ( ) | ( ) | ( ) | ( ) | ( ) | Strongly agree |

5. **I respond to smartphone notifications regularly.** *
 *Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ( ) | ( ) | ( ) | ( ) | ( ) | ( ) | ( ) | Strongly agree |

6. **When you receive a notification, how long do you usually wait to look at it? ***
*Mark only one oval.*

- ◯ I usually stop what I am doing and look to see if it is important
- ◯ I usually wait until a pre-defined break in the day (e.g. lunchtime) and then look
- ◯ I usually finish my immediate task (e.g. typing an email) and then look

## Typing

7. **I type on a keyboard at a desktop pc or laptop regularly. ***
*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

8. **When you are typing at a desktop pc or laptop, where would you usually leave your mobile device? ***
*Tick all that apply.*

- ☐ Beside me (Left or Right)
- ☐ In my trouser pocket
- ☐ In my jacket pocket
- ☐ In my bag
- ☐ In front of me
- ☐ In another room
- ☐ Other: ____________________

9. **Of the following, which would you use to read your smartphone notifications while typing at a computer? ***
*Mark only one oval.*

- ◯ Lockscreen
- ◯ Pop-up on a Desktop PC or laptop
- ◯ Smart watch (e.g. Android wear, pebble)
- ◯ Situated display
- ◯ Smart eyewear (e.g. Google Glass)
- ◯ Notification bar

# Notification Displays Experiment

Please complete the following questions about the task that you have just completed. There are 8 questions in total and they should take no longer than 2 minutes to complete.

*Required

1. **P id** *

   Completed by experimenter

   ..............................................................

## Notification Display

2. **It was easy to interpret the content of a notification using this display method.** *

   *Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

3. **It was easy to concentrate on the typing task using this display method.** *

   *Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

4. **It was convenient to use this display to read the notification.** *

   *Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

5. **It was frustrating to use this display to read the notification.** *

   *Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

6. **The notifications felt urgent when I used this display.** *

*Mark only one oval.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
|---|---|---|---|---|---|---|---|---|
| Strongly disagree | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ | Strongly agree |

## User Experience

7. **What did you like about using this display method?** *

..........................................................................

8. **What did you not like about using this display method?** *

..........................................................................

9. **Are there any situations that you could imagine using this display method?** *

..........................................................................

# Notifications Experiment: Final Questionnaire

Please complete the following questions about the experiment that you have just completed.
There are 2 questions in total and they should take no longer than 2 minutes to complete.

*Required

1. **P id** *
   Completed by experimenter

   ..............................................................................

## Overall

2. **Please rank in order of preference** *
   *Mark only one oval per row.*

| | 1 = MOST preferred | 2 | 3 | 4 | 5 | 6 = LEAST preferred |
|---|---|---|---|---|---|---|
| Lockscreen | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ |
| Smart watch | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ |
| Desktop pop-up | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ |
| Notification bar | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ |
| Situated display | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ |
| Smart Eyewear | ◯ | ◯ | ◯ | ◯ | ◯ | ◯ |

## Final comments

3. **Do you have any final comments about this experiment?** *

   ..............................................................................

# Bibliography

[1] ADOMAVICIUS, G., AND TUZHILIN, A. Context-aware recommender systems. In *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Springer US, 2011, pp. 217–253.

[2] AH KUN, L. M., AND MARSDEN, G. Co-present photo sharing on mobile devices. In *Proceedings of the 9th International Conference on Human Computer Interaction with Mobile Devices and Services* (New York, NY, USA, 2007), MobileHCI '07, ACM, pp. 277–284.

[3] AILISTO, H., POHJANHEIMO, L., VÄLKKYNEN, P., STRÖMMER, E., TUOMISTO, T., AND KORHONEN, I. Bridging the physical and virtual worlds by local connectivity-based physical selection. *Personal and Ubiquitous Computing* (2006), 333–344.

[4] ASLAN, I., MURER, M., PRIMESSNIG, F., MOSER, C., AND TSCHELIGI, M. The digital bookshelf: Decorating with collections of digital books. In *Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication* (New York, NY, USA, 2013), UbiComp '13 Adjunct, ACM, pp. 777–784.

[5] AZZOPARDI, L. Modelling interaction with economic models of search. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval* (New York, NY, USA, 2014), SIGIR '14, ACM, pp. 3–12.

[6] BAHL, P., AND PADMANABHAN, V. N. Radar: An in-building RF-based user location and tracking system. In *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies* (2000), vol. 2 of *INFOCOM '00*, IEEE, pp. 775–784.

[7] BALLENDAT, T., MARQUARDT, N., AND GREENBERG, S. Proxemic interaction: Designing for a proximity and orientation-aware environment. In *ACM International Conference on Interactive Tabletops and Surfaces* (New York, NY, USA, 2010), ITS '10, ACM, pp. 121–130.

[8] BARGH, M. S., AND DE GROOTE, R. Indoor localization based on response rate of bluetooth inquiries. In *Proceedings of the First ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments* (New York, NY, USA, 2008), MELT '08, ACM, pp. 49–54.

[9] BEKKELIEN, A. Bluetooth indoor positioning. *Master of Computer Science, University of Geneva* (2012), 1–49.

[10] BEKKELIEN, A., AND DERIAZ, M. Hybrid positioning framework for mobile devices. In *Ubiquitous Positioning, Indoor Navigation, and Location Based Service (UPINLBS), 2012* (2012), IEEE, pp. 1–7.

[11] BELLOTTI, V., AND EDWARDS, K. Intelligibility and accountability: Human considerations in context-aware systems. *Human-Computer Interaction 16*, 2 (Dec. 2001), 193–212.

[12] BENYON, D., MIVAL, O., AND AYAN, S. Designing blended spaces. In *Proceedings of the 26th Annual BCS Interaction Specialist Group Conference on People and Computers* (Swinton, UK, UK, 2012), BCS-HCI '12, British Computer Society, pp. 398–403.

[13] BERGSTROM-LEHTOVIRTA, J., OULASVIRTA, A., AND BREWSTER, S. The effects of walking speed on target acquisition on a touchscreen interface. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services* (New York, NY, USA, 2011), MobileHCI '11, ACM, pp. 143–146.

[14] BILLINGHURST, M., BOWSKILL, J., DYER, N., AND MORPHETT, J. Spatial information displays on a wearable computer. *Computer Graphics and Applications, IEEE* (1998), 24–31.

[15] BÖHMER, M., GANEV, L., AND KRÜGER, A. Appfunnel: A framework for usage-centric evaluation of recommender systems that suggest mobile applications. IUI '13.

[16] BÖHMER, M., HECHT, B., SCHÖNING, J., KRÜGER, A., AND BAUER, G. Falling asleep with angry birds, facebook and kindle: a large scale study on mobile application usage. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services* (New York, NY, USA, 2011), MobileHCI '11, ACM, pp. 47–56.

[17] BÖHMER, M., AND KRÜGER, A. Gaming the android OS for improving the design of smartphone launchers. Workshop on Informing Future Design via Large-Scale Research Methods and Big Data, at MobileHCI '13.

[18] BÖHMER, M., AND KRÜGER, A. A study on icon arrangement by smartphone users. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2013), CHI '13, ACM, pp. 2137–2146.

[19] BÖHMER, M., SAPONAS, T. S., AND TEEVAN, J. Smartphone use does not have to be rude: Making phones a collaborative presence in meetings. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services* (New York, NY, USA, 2013), MobileHCI '13, ACM, pp. 342–351.

[20] BREHMER, M., GRAHAM, T. C. N., AND STACH, T. Activate your gaim: A toolkit for input in active games. In *Proceedings of the International Academic Conference on the Future of Game Design and Technology* (New York, NY, USA, 2010), FuturePlay '10, ACM, pp. 151–158.

[21] BREWSTER, S. Overcoming the lack of screen space on mobile computers. *Personal Ubiquitous Comput. 6*, 3 (Jan. 2002), 188–205.

[22] BUCHANAN, G. R. The fused library: Integrating digital and physical libraries with location-aware sensors. In *Proceedings of the 10th Annual Joint Conference on Digital Libraries* (2010), JCDL '10, pp. 273–282.

[23] BUDZIK, J., AND HAMMOND, K. J. User interactions with everyday applications as context for just-in-time information access. In *Proceedings of the 5th International Conference on Intelligent User Interfaces* (New York, NY, USA, 2000), IUI '00, ACM, pp. 44–51.

[24] CAON, M., YUE1, Y., TSCHERRIG, J., MUGELLINI, E., AND KHALED, O. A. Context-aware 3d gesture interaction based on multiple kinects. *AMBIENT '11* (2011), 7–12.

[25] CHALMERS, M., AND GALANI, A. Seamful interweaving: Heterogeneity in the theory and design of interactive systems. In *Proceedings of the 5th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques* (New York, NY, USA, 2004), DIS '04, ACM, pp. 243–252.

[26] CHAWATHE, S. Beacon placement for indoor localization using bluetooth. In *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on* (Oct 2008), pp. 980–985.

[27] CHEOK, A. D., AND LI, Y. U. Ubiquitous interaction with positioning and navigation using a novel light sensor-based information transmission system. *Personal Ubiquitous Comput* (2008), 445–458.

[28] Cherubini, M., De Oliveira, R., Hiltunen, A., and Oliver, N. Barriers and bridges in the adoption of today's mobile phone contextual services. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services* (2011), ACM, pp. 167–176.

[29] Cheung, K., Intille, S., and Larson, K. An inexpensive bluetooth-based indoor location hack. In *Extended Abstracts UbiComp* (2006), pp. 1–2.

[30] Church, K., and Smyth, B. Understanding the intent behind mobile information needs. In *Proceedings of the 14th International Conference on Intelligent User Interfaces* (New York, NY, USA, 2009), IUI '09, ACM, pp. 247–256.

[31] Cockburn, A., Gutwin, C., and Greenberg, S. A predictive model of menu performance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2007), CHI '07, ACM, pp. 627–636.

[32] Cooperstock, J. R., Fels, S. S., Buxton, W., and Smith, K. C. Reactive environments. *Commun. ACM 40*, 9 (Sept. 1997), 65–73.

[33] Cornejo, R., Favela, J., and Tentori, M. Ambient displays for integrating older adults into social networking sites. In *Collaboration and Technology*, G. Kolfschoten, T. Herrmann, and S. Lukosch, Eds., vol. 6257 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2010, pp. 321–336.

[34] Costanza, E., Inverso, S. A., Pavlov, E., Allen, R., and Maes, P. Eye-q: Eyeglass peripheral display for subtle intimate notifications. In *Proceedings of the 8th Conference on Human-computer Interaction with Mobile Devices and Services* (New York, NY, USA, 2006), MobileHCI '06, ACM, pp. 211–218.

[35] Crowley, T., Milazzo, P., Baker, E., Forsdick, H., and Tomlinson, R. Mmconf: An infrastructure for building shared multimedia applications. In *Proceedings of the 1990 ACM Conference on Computer-supported Cooperative Work* (New York, NY, USA, 1990), CSCW '90, ACM, pp. 329–342.

[36] Dahley, A., Wisneski, C., and Ishii, H. Water lamp and pinwheels: Ambient projection of digital information into architectural space. In *CHI 98 Cconference Summary on Human Factors in Computing Systems* (New York, NY, USA, 1998), CHI '98, ACM, pp. 269–270.

[37] Davidsson, C., and Moritz, S. Utilizing implicit feedback and context to recommend mobile applications from first use. In *Proceedings of the 2011 Workshop on Context-awareness in Retrieval and Recommendation* (New York, NY, USA, 2011), CaRR '11, ACM, pp. 19–22.

[38] DEARMAN, D., KELLAR, M., AND TRUONG, K. N. An examination of daily information needs and sharing opportunities. In *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work* (New York, NY, USA, 2008), CSCW '08, ACM, pp. 679–688.

[39] DEY, A. K. Understanding and using context. *Personal Ubiquitous Computing 5*, 1 (Jan. 2001), 4–7.

[40] DEY, A. K., WAC, K., FERREIRA, D., TASSINI, K., HONG, J.-H., AND RAMOS, J. Getting closer: An empirical investigation of the proximity of user to their smart phones. In *Proceedings of the 13th International Conference on Ubiquitous Computing* (New York, NY, USA, 2011), UbiComp '11, ACM, pp. 163–172.

[41] DIX, A., RODDEN, T., DAVIES, N., TREVOR, J., FRIDAY, A., AND PALFREYMAN, K. Exploiting space and location as a design framework for interactive mobile systems. *ACM Trans. Comput.-Hum. Interact.* (2000), 285–321.

[42] DOURISH, P. What we talk about when we talk about context. *Personal Ubiquitous Comput. 8*, 1 (2004), 19–30.

[43] DREWNIAK, P. Augmenting mobile interaction with ambient lighting. *Master in Computer Science, University of Glasgow* (2014), 1–10.

[44] EAGLE, N., AND PENTLAND, A. Reality mining: sensing complex social systems. *Personal Ubiquitous Comput. 10*, 4 (May 2006), 255–268.

[45] EVANS, J. *Straightforward Statistics for the Behavioral Sciences*. Duxbury Press, 1996.

[46] FAGIN, R., KUMAR, R., AND SIVAKUMAR, D. Comparing top k lists. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms* (2003), SODA '03, Society for Industrial and Applied Mathematics, pp. 28–36.

[47] FALAKI, H., MAHAJAN, R., KANDULA, S., LYMBEROPOULOS, D., GOVINDAN, R., AND ESTRIN, D. Diversity in smartphone usage. MobiSys '10, pp. 179–194.

[48] FELDMANN, S., KYAMAKYA, K., ZAPATER, A., AND LUE, Z. An Indoor Bluetooth-Based Positioning System: Concept, Implementation and Experimental Evaluation. In *International Conference on Wireless Networks* (2003), pp. 109–113.

[49] FINDLATER, L., AND MCGRENERE, J. A comparison of static, adaptive, and adaptable menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2004), CHI '04, ACM, pp. 89–96.

[50] FINDLATER, L., AND MCGRENERE, J. Impact of screen size on performance, awareness, and user satisfaction with adaptive graphical user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2008), CHI '08, ACM, pp. 1247–1256.

[51] FINDLATER, L., MOFFATT, K., MCGRENERE, J., AND DAWSON, J. Ephemeral adaptation: The use of gradual onset to improve menu selection performance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2009), CHI '09, pp. 1655–1664.

[52] FITCHETT, S., AND COCKBURN, A. AccessRank: Predicting what users will do next. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2012), CHI '12, ACM, pp. 2239–2242.

[53] FITZMAURICE, G., AND BUXTON, W. The chameleon: spatially aware palmtop computers. CHI '94, pp. 451–452.

[54] FITZMAURICE, G. W. Situated information spaces and spatially aware palmtop computers. *Commun. ACM 36* (1993), 39–49.

[55] FLEMISCH, F. O., ADAMS, C. A., CONWAY, S. R., GOODRICH, K. H., PALMER, M. T., AND SCHUTTE, P. C. The H-metaphor as a guideline for vehicle automation and interaction. *Technical Report NASA* (2003).

[56] FONSECA, H. C., NEVES, A. R. D. M., AND RALHA, C. G. A user location case study using different wireless protocols. MobiWac '11, pp. 143–146.

[57] FUKAZAWA, Y., HARA, M., ONOGI, M., AND UENO, H. Automatic mobile menu customization based on user operation history. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services* (2009), ACM, p. 50.

[58] GAJOS, K. Z., CZERWINSKI, M., TAN, D. S., AND WELD, D. S. Exploring the design space for adaptive graphical user interfaces. In *Proceedings of the Working Conference on Advanced Visual Interfaces* (New York, NY, USA, 2006), AVI '06, ACM, pp. 201–208.

[59] GAJOS, K. Z., EVERITT, K., TAN, D. S., CZERWINSKI, M., AND WELD, D. S. Predictability and accuracy in adaptive user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2008), CHI '08, ACM, pp. 1271–1274.

[60] GEHRING, S., LÖCHTEFELD, M., DAIBER, F., BÖHMER, M., AND KRÜGER, A. Using intelligent natural user interfaces to support sales conversations. 97–100.

[61] GREENHALGH, C., FRENCH, A., TENNENT, P., HUMBLE, J., AND CRABTREE, A. From ReplayTool to digital replay system. In *Proceedings of the 3rd International Conference on e-Social Science* (October 2007), pp. 1–10.

[62] GRIMES, A., TARASEWICH, P., AND CAMPBELL, C. Keeping information private in the mobile environment. In *Proceedings of the 2005 Workshop on Social Implications of Ubiquitous Computing at CHI '05* (2005), CHI '05.

[63] GUSTAFSON, S., BIERWIRTH, D., AND BAUDISCH, P. Imaginary interfaces: spatial interaction with empty hands and without visual feedback. UIST '10, pp. 3–12.

[64] GUSTAFSON, S., HOLZ, C., AND BAUDISCH, P. Imaginary phone: learning imaginary interfaces by transferring spatial memory from a familiar device. In *Proceedings of the 24th annual ACM symposium on User interface software and technology* (New York, NY, USA, 2011), UIST '11, ACM, pp. 283–292.

[65] HANG, A., BROLL, G., AND WIETHOFF, A. Visual design of physical user interfaces for NFC-based mobile interaction. *DIS '10* (2010), 292–301.

[66] HANG, A., DE LUCA, A., HARTMANN, J., AND HUSSMANN, H. Oh app, where art thou?: On app launching habits of smartphone users. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services* (New York, NY, USA, 2013), MobileHCI '13, ACM, pp. 392–395.

[67] HANSSON, R., AND LJUNGSTRAND, P. The reminder bracelet: Subtle notification cues for mobile devices. In *CHI '00 Extended Abstracts on Human Factors in Computing Systems* (New York, NY, USA, 2000), CHI EA '00, ACM, pp. 323–324.

[68] HANSSON, R., LJUNGSTRAND, P., AND REDSTRÖM, J. Subtle and public notification cues for mobile devices. In *Proceedings of the 3rd International Conference on Ubiquitous Computing* (London, UK, UK, 2001), UbiComp '01, Springer-Verlag, pp. 240–246.

[69] HARDY, R., RUKZIO, E., HOLLEIS, P., AND WAGNER, M. Mobile interaction with static and dynamic NFC-based displays. 123–132.

[70] HARDY, R., RUKZIO, E., HOLLEIS, P., AND WAGNER, M. MyState: sharing social and contextual information through touch interactions with tagged objects. *MobileHCI '11* (2011), 475–484.

[71] HARPER, R. *The Connected Home: The Future of Domestic Life*. Springer, 2011.

[72] HARPER, R., RODDEN, T., ROGERS, Y., AND SELLEN, A. Being human: HCI in the year 2020. *Microsoft Research* (2008), 32–51.

[73] HARR, R., AND KAPTELININ, V. Unpacking the social dimension of external interruptions. In *Proceedings of the 2007 International ACM Conference on Supporting Group Work* (New York, NY, USA, 2007), GROUP '07, ACM, pp. 399–408.

[74] HARTER, A., HOPPER, A., STEGGLES, P., WARD, A., AND WEBSTER, P. The anatomy of a context-aware application. In *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking* (New York, NY, USA, 1999), MobiCom '99, ACM, pp. 59–68.

[75] HASSENZAHL, M. The interplay of beauty, goodness, and usability in interactive products. *Hum.-Comput. Interact. 19*, 4 (Dec. 2008), 319–349.

[76] HAY, S., AND HARLE, R. Bluetooth tracking without discoverability. In *Location and Context Awareness*, T. Choudhury, A. Quigley, T. Strang, and K. Suginuma, Eds., vol. 5561 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2009, pp. 120–137.

[77] HEIMONEN, T. Information needs and practices of active mobile internet users. Mobility '09, pp. 50:1–50:8.

[78] HOGGAN, E., AND BREWSTER, S. Designing audio and tactile crossmodal icons for mobile devices. ICMI '07, pp. 162–169.

[79] HOIS, J. *Knowledge Science, Engineering and Management: 4th International Conference, KSEM 2010, Belfast, Northern Ireland, UK, September 1-3, 2010. Proceedings*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, ch. Modularizing Spatial Ontologies for Assisted Living Systems, pp. 424–435.

[80] HORVITZ, E. Machine learning, reasoning and intelligence in daily life. *Technical Report TR-2006-185, Microsoft Research* (2006).

[81] HOSSAIN, A., AND SOH, W.-S. A comprehensive study of bluetooth signal parameters for localization. In *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on* (Sept 2007), pp. 1–5.

[82] HUI, B., PARTRIDGE, G., AND BOUTILIER, C. A probabilistic mental model for estimating disruption. In *Proceedings of the 14th international conference on Intelligent user interfaces* (New York, NY, USA, 2009), IUI '09, ACM, pp. 287–296.

[83] ISHII, H., AND ULLMER, B. Tangible bits: Towards seamless interfaces between people, bits and atoms. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 1997), CHI '97, ACM, pp. 234–241.

[84] KARATZOGLOU, A., BALTRUNAS, L., CHURCH, K., AND BÖHMER, M. Climbing the app wall: enabling mobile app discovery through context-aware recommendations. CIKM '12, pp. 2527–2530.

[85] KARIKOSKI, J., AND SOIKKELI, T. Contextual usage patterns in smartphone communication services. *Personal Ubiquitous Comput. 17*, 3 (Mar. 2013), 491–502.

[86] KAUFMANN, B., AND BUECHLEY, L. Amarino: a toolkit for the rapid prototyping of mobile ubiquitous computing. *MobileHCI '10* (2010), 291–298.

[87] KIRSH, D. The Intelligent Use of Space. *Artificial Intelligence 73*, 1-2 (1995), 31–68.

[88] KNUTH, D. E. Computer programming as an art. *Commun. ACM 17, 12 '74* (1974), 667–673.

[89] KRANZ, M., HOLLEIS, P., AND SCHMIDT, A. Embedded interaction: Interacting with the internet of things. *IEEE Internet Computing* (2010), 46–53.

[90] KUFLIK, T., LANIR, J., DIM, E., WECKER, A., CORRA', M., ZANCANARO, M., AND STOCK, O. Indoor positioning: challenges and solutions for indoor cultural heritage sites. *IUI '11* (2011), 375–378.

[91] LAMARCA, A., CHAWATHE, Y., CONSOLVO, S., HIGHTOWER, J., SMITH, I., SCOTT, J., SOHN, T., HOWARD, J., HUGHES, J., POTTER, F., TABERT, J., POWLEDGE, P., BORRIELLO, G., AND SCHILIT, B. Place lab: Device positioning using radio beacons in the wild. In *Proceedings of the Third International Conference on Pervasive Computing* (Berlin, Heidelberg, 2005), PERVASIVE'05, Springer-Verlag, pp. 116–133.

[92] LEE, S., TANIN, E., AND KULIK, L. A simple localization system for ad-hoc indoor meetings through wireless connection points. In *Proceedings of the 13th International Conference on Mobile and Ubiquitous Multimedia* (New York, NY, USA, 2014), MUM '14, ACM, pp. 220–223.

[93] LI, F. C. Y., DEARMAN, D., AND TRUONG, K. N. Virtual shelves: Interactions with orientation aware devices. In *Proceedings of the 22Nd Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, 2009), UIST '09, ACM, pp. 125–128.

[94] LI, I., DEY, A. K., AND FORLIZZI, J. Understanding my data, myself: Supporting self-reflection with ubicomp technologies. In *Proceedings of the 13th International Conference on Ubiquitous Computing* (New York, NY, USA, 2011), UbiComp '11, ACM, pp. 405–414.

[95] LIM, B. Y., AND DEY, A. K. Investigating intelligibility for uncertain context-aware applications. In *Proceedings of the 13th International Conference on Ubiquitous Computing* (New York, NY, USA, 2011), UbiComp '11, ACM, pp. 415–424.

[96] LINDQVIST, J., CRANSHAW, J., WIESE, J., HONG, J., AND ZIMMERMAN, J. I'm the mayor of my house: examining why people use foursquare - a social-driven location sharing application. In *Proc. of 2011 Human factors* (2011), CHI '11, pp. 2409–2418.

[97] LITTLE, L., AND BRIGGS, P. Private whispers/public eyes: Is receiving highly personal information in a public place stressful? *Interact. Comput. 21*, 4 (Aug. 2009), 316–322.

[98] LÖCHTEFELD, M., BÖHMER, M., AND GANEV, L. AppDetox: Helping users with mobile app addiction. In *Proceedings of the 12th International Conference on Mobile and Ubiquitous Multimedia* (New York, NY, USA, 2013), MUM '13, ACM, pp. 43:1–43:2.

[99] LUCERO, A., HOLOPAINEN, J., AND JOKELA, T. Pass-them-around: Collaborative use of mobile phones for photo sharing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2011), CHI '11, ACM, pp. 1787–1796.

[100] LUCERO, A., AND VETEK, A. Notifeye: Using interactive glasses to deal with notifications while walking in public. In *Proceedings of the 11th Conference on Advances in Computer Entertainment Technology* (New York, NY, USA, 2014), ACE '14, ACM, pp. 17:1–17:10.

[101] MACKAY, W. E., AND FAYARD, A.-L. HCI, natural science and design: A framework for triangulation across disciplines. In *Proceedings of the 2nd Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques* (New York, NY, USA, 1997), DIS '97, ACM, pp. 223–234.

[102] MACKENZIE, I. S., AND SOUKOREFF, R. W. Phrase sets for evaluating text entry techniques. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems* (New York, NY, USA, 2003), CHI EA '03, ACM, pp. 754–755.

[103] MADHAVAPEDDY, A., AND TSE, A. A study of bluetooth propagation using accurate indoor location mapping. In *Proceedings of the 7th International Conference on Ubiquitous Computing* (Berlin, Heidelberg, 2005), UbiComp'05, Springer-Verlag, pp. 105–122.

[104] MARIAKAKIS, A. T., SEN, S., LEE, J., AND KIM, K.-H. Sail: Single access point-based indoor localization. In *Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services* (New York, NY, USA, 2014), MobiSys '14, ACM, pp. 315–328.

[105] MARQUARDT, N., DIAZ-MARINO, R., BORING, S., AND GREENBERG, S. The proximity toolkit: Prototyping proxemic interactions in ubiquitous computing ecologies. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, 2011), UIST '11, ACM, pp. 315–326.

[106] MCFARLANE, D. Comparison of four primary methods for coordinating the interruption of people in human-computer interaction. *Hum.-Comput. Interact. 17*, 1 (Mar. 2002), 63–139.

[107] MCFARLANE, D. C. Interruption of people in human-computer interaction: A general unifying definition of human interruption and taxonomy. Tech. rep., Technical Report NRL/FR/5510-97-9870, US Naval Research Lab, Washington, DC., 1997.

[108] MORRISON, A., MCMILLAN, D., REEVES, S., SHERWOOD, S., AND CHALMERS, M. A hybrid mass participation approach to mobile software trials. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2012), CHI '12, ACM, pp. 1311–1320.

[109] NI, L. M., LIU, Y., LAU, Y. C., AND PATIL, A. P. LANDMARC: Indoor location sensing using active RFID. *Wireless Networks - Special issue: Pervasive computing and communications 10*, 6 (Nov. 2004), 701–710.

[110] NORRIE, L., KOELLE, M., MURRAY-SMITH, R., AND KRANZ, M. Putting books back on the shelf: Situated interactions with digital book collections on smartphones. In *Proceedings of the 12th International Conference on Mobile and Ubiquitous Multimedia* (New York, NY, USA, 2013), MUM '13, ACM, pp. 44:1–44:2.

[111] NORRIE, L., AND MURRAY-SMITH, R. Interacting with multiple mobile devices using the kinect. *Workshop on Mobile Gestures at Mobile HCI '11* (2011), 1–4.

[112] NORRIE, L., AND MURRAY-SMITH, R. Virtual Sensors: Rapid prototyping of ubiquitous interaction with a mobile phone and a kinect. Mobile HCI '11, pp. 25–28.

[113] NORRIE, L., AND MURRAY-SMITH, R. Cast Together: Inclusive and unobtrusive mobile interactions with a situated display. In *Proceedings of the 4th International Symposium on Pervasive Displays* (2015), ACM, pp. 237–238.

[114] NORRIE, L., AND MURRAY-SMITH, R. Impact of smartphone notification display choice in a typing task. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct* (2015), ACM, pp. 1094–1099.

[115] NORRIE, L., AND MURRAY-SMITH, R. Investigating UI displacements in an adaptive mobile homescreen. *International Journal of Mobile Human Computer Interaction 8*, 3 (2015), 1–18.

[116] NORRIE, L., AND MURRAY-SMITH, R. Notification display choice for smartphone users: Investigating the impact of notification displays on a typing task. *International Journal of Mobile Human Computer Interaction 8*, 5 (2016), 1–23.

[117] O'HARA, K., LIPSON, M., JANSEN, M., UNGER, A., JEFFRIES, H., AND MACER, P. Jukola: Democratic music choice in a public space. In *Proceedings of the 5th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques* (New York, NY, USA, 2004), DIS '04, ACM, pp. 145–154.

[118] OLSSON, T., AND SALO, M. Narratives of satisfying and unsatisfying experiences of current mobile augmented reality applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2012), CHI '12, ACM, pp. 2779–2788.

[119] PEARSON, J., ROBINSON, S., AND JONES, M. It's about time: Smartwatches as public displays. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (New York, NY, USA, 2015), CHI '15, ACM, pp. 1257–1266.

[120] PIELOT, M., CHURCH, K., AND DE OLIVEIRA, R. An in-situ study of mobile phone notifications. In *Proceedings of the 16th International Conference on Human-Computer Interaction with Mobile Devices & Services* (New York, NY, USA, 2014), MobileHCI '14, ACM, pp. 233–242.

[121] PRIYANTHA, N. B., CHAKRABORTY, A., AND BALAKRISHNAN, H. The cricket location-support system. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking* (New York, NY, USA, 2000), MobiCom '00, ACM, pp. 32–43.

[122] RAHMAN, A. S. M. M., HOSSAIN, M. A., AND SADDIK, A. E. Spatial-geometric approach to physical mobile interaction based on accelerometer and IR sensory data

fusion. *ACM Transactions on Multimedia Computing, Communications and Appliances* (2010), 28:1–28:23.

[123] RANDELL, C., AND MULLER, H. L. Low cost indoor positioning system. *UbiComp '01* (2001), 42–48.

[124] RASHID, U., NACENTA, M. A., AND QUIGLEY, A. Factors influencing visual attention switch in multi-display user interfaces: A survey. In *Proceedings of the 2012 International Symposium on Pervasive Displays* (New York, NY, USA, 2012), PerDis '12, ACM, pp. 1:1–1:6.

[125] RAWASSIZADEH, R., TOMITSCH, M., WAC, K., AND TJOA, A. M. UbiqLog: A generic mobile phone-based life-log framework. *Personal Ubiquitous Comput. 17*, 4 (Apr. 2013), 621–637.

[126] REEVES, S., BENFORD, S., O'MALLEY, C., AND FRASER, M. Designing the spectator experience. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2005), CHI '05, ACM, pp. 741–750.

[127] RHODES, B. J. The wearable remembrance agent: A system for augmented memory. In *Proceedings of the 1st IEEE International Symposium on Wearable Computers* (Washington, DC, USA, 1997), ISWC '97, IEEE Computer Society, pp. 123–.

[128] ROGERS, Y. Moving on from Weiser's vision of calm computing: Engaging ubiComp experiences. *Ubicomp '06* (2006), 404–421.

[129] ROSSI, M., SEITER, J., AMFT, O., BUCHMEIER, S., AND TRÖSTER, G. Roomsense: An indoor positioning system for smartphones using active sound probing. In *Proceedings of the 4th Augmented Human International Conference* (New York, NY, USA, 2013), AH '13, ACM, pp. 89–95.

[130] ROUNCEFIELD, M., AND TOLMIE, P. *Chapter 8 of The Connected Home: The Future of Domestic Life*. Springer London, London, 2011.

[131] RUKZIO, E., SCHMIDT, A., HUSSMANN, H., AND INFORMATICS, M. An analysis of the usage of mobile phones for personalized interactions with ubiquitous public displays. workshop ubiquitous display environments, ubicomp 2004. In *Ubiquitous Public Displays. Workshop on Ubiquitous Display Environments* (2004), UbiComp 2004.

[132] SAHAMI SHIRAZI, A., HENZE, N., DINGLER, T., PIELOT, M., WEBER, D., AND SCHMIDT, A. Large-scale assessment of mobile notifications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2014), CHI '14, ACM, pp. 3055–3064.

[133] SCARR, J., COCKBURN, A., GUTWIN, C., AND MALACRIA, S. Testing the robustness and performance of spatially consistent interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2013), CHI '13, ACM, pp. 3139–3148.

[134] SCHWARZ, J., KLIONSKY, D., HARRISON, C., DIETZ, P., AND WILSON, A. Phone as a pixel: Enabling ad-hoc, large-scale displays using mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2012), CHI '12, ACM, pp. 2235–2238.

[135] SEARS, A., AND SHNEIDERMAN, B. Split menus: Effectively using selection frequency to organize menus. *ACM Trans. Comput.-Hum. Interact. 1*, 1 (Mar. 1994), 27–51.

[136] SHABIB, N., AND KROGSTIE, J. The use of data mining techniques in location-based recommender system. In *Proceedings of the International Conference on Web Intelligence, Mining and Semantics* (2011), WIMS '11, pp. 28:1–28:7.

[137] SHERIDAN, S. *Humans and automation: System design and Research Issues*. Wiley, 2002.

[138] SHIN, C., HONG, J.-H., AND DEY, A. K. Understanding and prediction of mobile application usage for smartphones. UbiComp'12, pp. 173–182.

[139] SHOEMAKER, G., TSUKITANI, T., KITAMURA, Y., AND BOOTH, K. S. Body-centric interaction techniques for very large wall displays. *NordiCHI '10* (2010), 463–472.

[140] SHOEMAKER, G., TSUKITANI, T., KITAMURA, Y., AND BOOTH, K. S. Body-centric interaction techniques for very large wall displays. NordiCHI '10, pp. 463–472.

[141] SOHN, T., LI, K. A., GRISWOLD, W. G., AND HOLLAN, J. D. A diary study of mobile information needs. CHI '08, pp. 433–442.

[142] STRACHAN, S., AND MURRAY-SMITH, R. Bearing-based selection in mobile spatial interaction. *Personal Ubiquitous Comput. 13*, 4 (May 2009), 265–280.

[143] STRACHAN, S., MURRAY-SMITH, R., AND O'MODHRAIN, S. BodySpace: Inferring body pose for natural control of a music player. In *CHI '07 Extended Abstracts on Human Factors in Computing Systems* (New York, NY, USA, 2007), CHI EA '07, ACM, pp. 2001–2006.

[144] SUNDSTRÖM, P., TAYLOR, A. S., AND O'HARA, K. Sketching in software and hardware bluetooth as a design material. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services* (New York, NY, USA, 2011), MobileHCI '11, ACM, pp. 405–414.

[145] TAN, C. S. S., SCHÖNING, J., LUYTEN, K., AND CONINX, K. Informing intelligent user interfaces by inferring affective states from body postures in ubiquitous computing environments. In *Proceedings of the 2013 International Conference on Intelligent User Interfaces* (New York, NY, USA, 2013), IUI '13, ACM, pp. 235–246.

[146] TANG, L., YU, Z., ZHOU, X., WANG, H., AND BECKER, C. Supporting rapid design and evaluation of pervasive applications: Challenges and solutions. *Personal Ubiquitous Comput. 15*, 3 (Mar. 2011), 253–269.

[147] TAPIA, D., ALONSO, R., RODRIGUEZ, S., DE LA PRIETA, F., CORCHADO, J., AND BAJO, J. Implementing a real-time locating system based on wireless sensor networks and artificial neural networks to mitigate the multipath effect. *Information Fusion (FUSION '11)* (2011), 1–8.

[148] TARASEWICH, P., GONG, J., AND CONLAN, R. Protecting private data in public. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems* (New York, NY, USA, 2006), CHI EA '06, ACM, pp. 1409–1414.

[149] THUDT, A., HINRICHS, U., AND CARPENDALE, S. The bohemian bookshelf: Supporting serendipitous book discoveries through information visualization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2012), CHI '12, ACM, pp. 1461–1470.

[150] TOSSELL, C., KORTUM, P., RAHMATI, A., SHEPARD, C., AND ZHONG, L. Characterizing web use on smartphones. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2012), CHI '12, ACM, pp. 2769–2778.

[151] TSANDILAS, T., AND SCHRAEFEL, M. C. An empirical assessment of adaptation techniques. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems* (New York, NY, USA, 2005), CHI EA '05, ACM, pp. 2009–2012.

[152] VERA, R., OCHOA, S. F., AND ALDUNATE, R. G. Edips: an easy to deploy indoor positioning system to support loosely coupled mobile work. *Personal Ubiquitous Comput.* (2011), 365–376.

[153] VERKASALO, H. Contextual patterns in mobile service usage. *Personal Ubiquitous Comput.* (2009), 331–342.

[154] VERTANEN, K., AND KRISTENSSON, P. O. A versatile dataset for text entry evaluations based on genuine mobile emails. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services* (New York, NY, USA, 2011), MobileHCI '11, ACM, pp. 295–298.

[155] WACHS, J. P., KLSCH, M., STERN, H., BEN-GURION, EDAN, Y., AND BEN-GURION. Vision-based hand-gesture applications. *Commun. ACM* (2011), 60–71.

[156] WANG, R., ZHAO, F., LUO, H., LU, B., AND LU, T. Fusion of Wi-Fi and bluetooth for indoor localization. In *Proceedings of the 1st International Workshop on Mobile Location-based Service* (New York, NY, USA, 2011), MLBS '11, ACM, pp. 63–66.

[157] WANT, R., FISHKIN, K. P., GUJAR, A., AND HARRISON, B. L. Bridging physical and virtual worlds with electronic tags. CHI '99, pp. 370–377.

[158] WEBBER, W., MOFFAT, A., AND ZOBEL, J. A similarity measure for indefinite rankings. *ACM Trans. Inf. Syst. 28*, 4 (Nov. 2010), 20:1–20:38.

[159] WEISER, M. The computer for the 21st century. *Scientific American 3*, 3 (July 1991), 94–104.

[160] WEISER, M., AND BROWN, J. S. Designing calm technology. *Powergrid 1* (1996). http://www.ubiq.com/weiser/calmtech/calmtech.htm.

[161] WILSON, A. D. New interaction concepts by using the Wii remote. *Proc of HCI. Part II: Novel Interaction Methods and Techniques* (2009), 261–270.

[162] WIMMER, R., HOLLEIS, P., KRANZ, M., AND SCHMIDT, A. Thracker - Using Capacitive Sensing for Gesture Recognition. *ICDCS Workshops '06* (2006), 64–66.

[163] WISNESKI, C., ISHII, H., DAHLEY, A., GORBET, M. G., BRAVE, S., ULLMER, B., AND YARIN, P. Ambient displays: Turning architectural space into an interface between people and digital information. In *Proceedings of the First International Workshop on Cooperative Buildings, Integrating Information, Organization, and Architecture* (London, UK, UK, 1998), CoBuild '98, Springer-Verlag, pp. 22–32.

[164] XU, Q., ERMAN, J., GERBER, A., MAO, Z., PANG, J., AND VENKATARAMAN, S. Identifying diverse usage behaviors of smartphone apps. IMC '11, pp. 329–344.

[165] YEE, K.-P. Peephole displays: Pen interaction on spatially aware handheld computers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2003), CHI '03, ACM, pp. 1–8.

[166] YOON, S., LEE, S.-S., LEE, J.-M., AND LEE, K. Understanding notification stress of smartphone messenger app. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems* (New York, NY, USA, 2014), CHI EA '14, ACM, pp. 1735–1740.

[167] ZHANG, C., DING, X., CHEN, G., HUANG, K., MA, X., AND YAN, B. Nihao: A predictive smartphone application launcher. In *Mobile Computing, Applications, and Services*. 2013, pp. 294–313.

[168] ZHANG, Y., AND WANG, L. Some challenges for context-aware recommender systems. In *ICCSE 2010* (2010), pp. 362–365.