

Proxemic-Aware Controls: Designing Remote Controls for Ubiquitous Computing Ecologies

David Ledo, Saul Greenberg
Department of Computer Science
University of Calgary
Calgary, Alberta, Canada
{dledomai, saul}@ucalgary.ca

Nicolai Marquardt
UCL Interaction Centre
University College London
Gower Street, London, UK
n.marquardt@ucl.ac.uk

Sebastian Boring
Department of Computer Science
University of Copenhagen
Copenhagen, Denmark
sebastian.boring@diku.dk

ABSTRACT

Remote controls facilitate interactions at-a-distance with appliances. However, the complexity, diversity, and increasing number of digital appliances in ubiquitous computing ecologies make it increasingly difficult to: (1) *discover* which appliances are controllable; (2) *select* a particular appliance from the large number available; (3) *view* information about its status; and (4) *control* the appliance in a pertinent manner. To mitigate these problems we contribute *proxemic-aware controls*, which exploit the spatial relationships between a person's handheld device and all surrounding appliances to create a dynamic appliance control interface. Specifically, a person can discover and select an appliance by the way one orients a mobile device around the room, and then progressively view the appliance's status and control its features in increasing detail by simply moving towards it. We illustrate proxemic-aware controls of assorted appliances through various scenarios. We then provide a generalized conceptual framework that informs future designs of proxemic-aware controls.

Author Keywords

Mobile Interaction, ubiquitous computing, proxemic-interaction, control of appliances.

ACM Classification Keywords

H.5.2. Information interfaces and presentation (e.g., HCI): user interfaces - interaction styles.

INTRODUCTION

Traditional remote controls were invented to allow people to interact with appliances at a distance. While originally wired and constrained to large appliances, such as televisions and radios, further advances led to a proliferation of wireless controls for a myriad of appliances: from traditional appliances such as air conditioners, sound systems and media centers, to the new generation of digital appliances. Remote controls initially duplicated the controls on an appliance. However, *Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org. MobileHCI '15, August 25 - 28, 2015, Copenhagen, Denmark Copyright is held by the owner/author(s). Publication rights licensed to ACM.*

ACM 978-1-4503-3652-9/15/08...\$15.00

DOI: <http://dx.doi.org/10.1145/2785830.2785871>

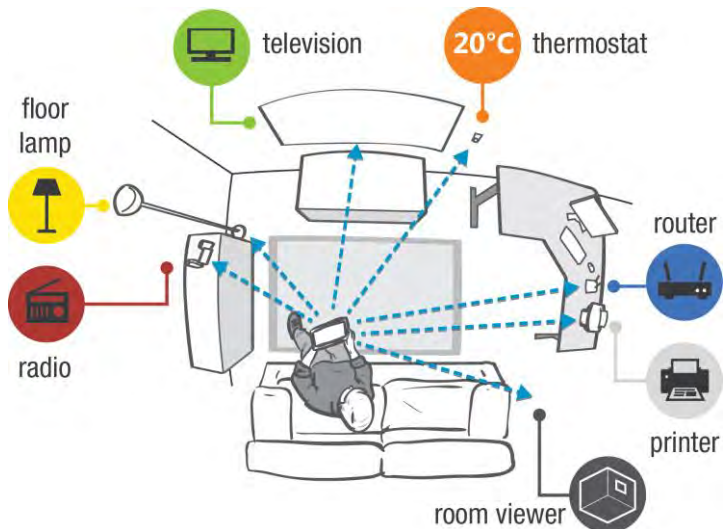


Figure 1. Mobile interaction with an ecology of appliances and devices, where a person has different spatial relationships with each of the interactive appliances in the room.

most contemporary remotes have become the primary interface to the appliance. This ‘off-loading’ of controls to the remote reduced costs and allowed for complex appliance functionality. Importantly, it also provided more design freedom to the appliance’s form factor (e.g., size, shape, materials, appearance) as large control panels no longer had to be embedded within it.

However, the increasing number of remotes led to scalability issues, as typified by the living room full of different remotes to control each individual appliance within it. To remedy this, *universal remotes* promoted a one-remote-to-many-appliances solution. Unfortunately, the universal remote introduced problems: it was often limited to entertainment systems, had difficult setup issues and poorly adaptable interfaces, and became yet another control joining a collection of already complex and inconsistent controls [32].

In 2002, Brad Myers advocated that the ubiquity and flexibility of personal mobile devices could serve as a suitable universal remote control to a new generation of digitally controllable appliances [18]. Since then, appliances have acquired the ability to interconnect and integrate themselves into a *ubiquitous computing ecology* [1] comprising the people and digital devices within a social space (Figure 1), e.g.,

a living room or a meeting room. As Myers predicted, such locations are increasingly equipped with a large number of appliances that can now be controlled with mobile devices.

However, new problems are emerging as the number of controllable appliances increases, as predicted by the Ubicomp and Internet of Things vision. First, it is difficult to *discover* at a glance which appliances are interactive. While the fixed appliances within a living room may be familiar to its family members, a meeting room with hidden projectors and speakers may require more intricate visual search by its temporary inhabitants. Once appliances are discovered, people still have to *select* an individual appliance from the large ecology. Once selected, people should be able to *view* information about the current status of the appliance, and progressively *control* its basic to advanced functions as needed without undue interface complexity.

To mitigate these problems we advocate *proxemic-aware controls*, which exploit the spatial relationships between a person's handheld device (serving as the universal remote) and all surrounding appliances to create a dynamic appliance control interface. Specifically, a person can discover and select an appliance by the way one orients a mobile device around the room, and then progressively view the appliance's status and control its features in increasing detail by simply moving towards it. This paper details the following contributions.

1. The notion of *proxemic-aware controls*, whose dynamic interface is based upon the spatial relationships between a person's handheld device and the surrounding appliances within an ubicomp ecology, is demonstrated through a series of implemented scenarios.
2. A *proxemic-aware control framework* that more generally informs the design of such controls, and that contextualizes prior literature within it.

PROXEMIC-AWARE CONTROLS

Proxemics is Edward Hall's seminal theory [8] about the way people use spatial relationships to mediate their interactions with other people around them. Hall observed how people continuously change and adapt their distance and orientation to others depending on social context and the task at hand. For example, we turn towards people we want to interact with, and move increasingly closer to them as a function of our relationship with them: from social, to personal, to intimate. Proxemics was later applied to ubicomp design, where *proxemic interactions* [1] introduced a first-order approximation of how sensed *proxemic variables* (distance, orientation, identity, movement, location) can be leveraged to mediate people's interactions with devices around them.

Our proxemic-aware controls are a particular class of proxemic-aware devices. They use the proxemic variables mentioned above to adapt a mobile control device's interface for interacting with appliances in the surrounding ubicomp environment. The spatial relationships – such as distance and

orientation – between the mobile device (acting as a universal controller) and the appliances directly adapt the interface content displayed and the controls offered to the user.

We considered several important goals when designing proxemic-aware remote controls for a ubicomp ecology.

1. ***Interactions should be situated in the physical world.*** In order to make appliance discovery and selection easy, the digital content shown on the remote control should be spatially associated to the physically present appliances. This is in direct contrast to interfaces that show a listing of all appliances known to it, regardless of whether or not those appliances are in the same physical location or room.
2. ***Interfaces should balance simplicity and flexibility of controls.*** When afar, people should be able to get a sense of the interactive appliances in the room as well as basic state information (e.g. its current primary settings). Controls can range from simple ones focused on basic tasks (e.g., turning something on/off); to rare or more complex operations (e.g., advanced settings, appliance configuration). This introduces a tradeoff between simplicity and flexibility [13]. As we will see, we use the notion of gradual engagement to seamlessly transition, as a function of proximity, from simple to complex controls. This is in line with Don Norman's studies and discussion on complexity, where people gain experience with tasks and progressively adjust to increasing levels of complexity [21].
3. ***Controls should enable seamless transition between appliances.*** This implies that the user should be able to quickly switch from controlling one appliance to selecting and controlling another appliance.
4. ***Proxemic-aware controls should complement existing approaches.*** Our goal is not to replace existing interaction paradigms for remote controls, such as pointing, touching or list selection (scanning). Instead, proxemic-aware controls should provide an alternative and complementary approach for interacting with appliances.

The next section illustrates seven scenarios of how proxemic-aware controls could work through a prototype that we built in our lab. A later section introduces our proxemic-aware controls framework, which discusses the types of appliances and the interaction models in further detail.

SCENARIOS FOR PROXEMIC-AWARE CONTROLS

We begin with an overview of our system and then describe seven implemented scenarios that illustrate the four design goals discussed above for proxemic-aware controls.

System overview. As shown in Figure 1, we created a home environment with six appliances (thermostat, floor lamp, radio, router, printer and a television) as a test-bed for demonstrating the expressiveness and versatility of proxemic-aware controls, and for exploring nuances of our design rationale. We built our system using the Proximity Toolkit [15] and a Vicon motion tracking system, which tracked the position of

a tablet (a Surface Pro 2) to the six appliances. Some of these appliances were custom-created physical appliances that can be digitally-controlled over a network (lamp, radio and television), while the others are digital simulations.

The remote control interface is realized on the tablet (an earlier version was built on a smart phone). The interface itself has several primary components, as annotated in Figure 2 and partially visible in Figure 3.

- **An overview** of discoverable appliances (as icons) is shown at the screen's edge. Each icon is in its correct spatial location relative to the appliances, where the icons reposition themselves as the tablet is moved. Two types of overviews are used: holding the tablet horizontally shows a bird's-eye overview (as in the Figure), whereas reorienting it vertically shows a panoramic overview.

- **The currently selected appliance** is shown at the screen's center as an interactive graphic. The graphic changes in size and in the amount of content presented as a function of proximity. As the person turns to another appliance, the current appliance animates out and the new one moves in.

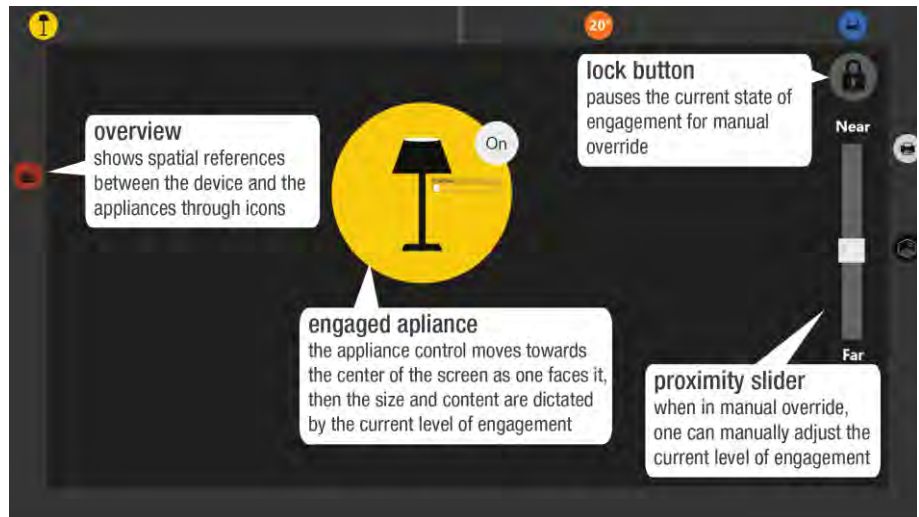


Figure 2. Interface for Proxemic-Aware Controls.

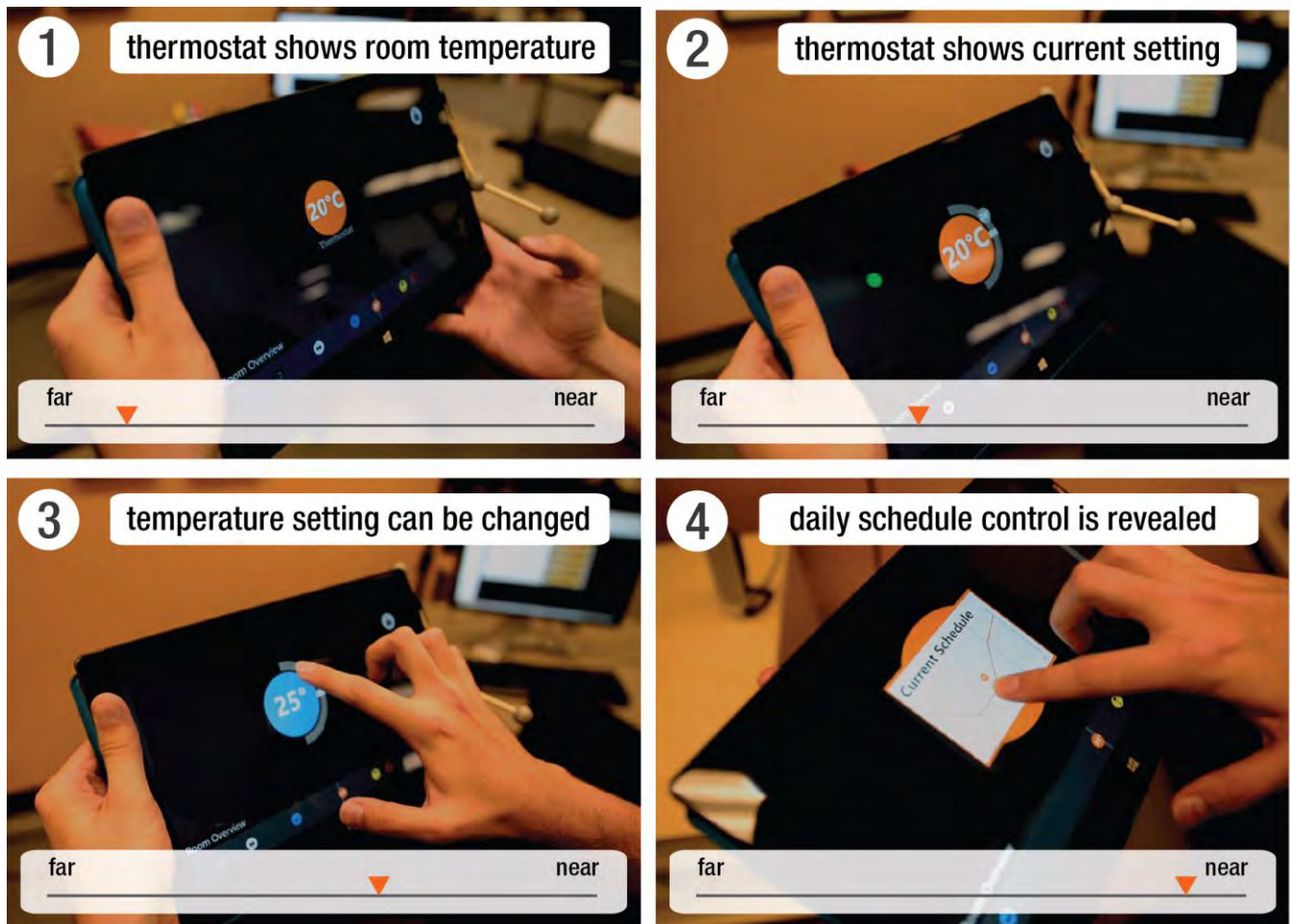


Figure 3. Gradually engaging with a thermostat – one can see different levels of information and controls as a function of physical proximity.

- **A Lock Button** is located at the top right corner. It pauses the spatial interaction to allow manual override.
- **A Proximity Slider** then appears below the Lock Button. When the locked, the person uses it to change the level of detail presented without actually having to move towards or away from the appliance, i.e., it acts as a surrogate to actual proximity.

Scenario 1: Discovering Interactive Appliances

Trevor walks into his living room. While looking at his tablet (Figure 1), he sees icons representing all of the appliances at the border, where the positions of the icons are animated to match their relative position to the physical appliances they represent (Figure 2, edges). He then rotates the tablet around the room to face each appliance: from the portable radio currently on the shelf, to the thermostat mounted on a wall, to a hidden router under the desk. As he does this, the appliance directly in front of the tablet is represented an interactive graphic in the center of the screen (Figure 2, center). While some appliances may have been moved since he was last in the room (e.g., the portable radio), the icons and the interactive graphic reflect the current appliance position.

This scenario describes how a proxemic-aware control makes it easy for a person to spatially scan a room. By moving the tablet, they can immediately see *what* appliances are part of the surrounding ubicomp ecology, and *where* they are located. Trevor can also choose *which* appliance he wants to interact with by simply facing it. All this occurs in moments of real time, where information is updated as a function of the person's proxemic relationship (orientation and distance) between the tablet and the surrounding appliances. All interactions are thus situated in the physical world.

Scenario 2: Gradual Engagement to an Appliance

Trevor feels a bit chilled. While facing his tablet towards the thermostat (which selects and shows it at the tablet's center), he sees the temperature of the room is currently 20°C (Figure 3.1 and Figure 4 top left). He moves closer to the thermostat, where its graphical control reveals (as a small labelled circle on the arc) that the thermostat is currently set to 22°C (Figure 3.2 and Figure 4 top, 2nd from left). As he continues his approach, that control becomes interactive, allowing him to increase the temperature setting (Figure 3.3 and Figure 4 top 3rd from left). However, he decides to check the thermostat's daily schedule – an advanced feature. He moves directly in front of the thermostat, and the heating schedule control appears (Figure 3.4 and Figure 4 top right). He decides to change it. He

locks the screen so he can move his tablet around without losing content, and changes the schedule by adjusting the schedule's control points.

This scenario illustrates how gradual engagement of controls [14] works as a function of proximity to provide a balance between simplicity and flexibility of controls. While this scenario focuses on a particular appliance (the thermostat), all other appliances implement this gradual engagement in a similar manner. Figure 4 shows how the interface to four appliances shows more detail at decreasing distances. By orienting his device towards the thermostat, Trevor was able to select it. The interface then uses semantic zoom: as Trevor moves towards the thermostat, his remote shows progressively more information of the thermostat state and creates opportunities for interaction (Figure 3 and Figure 4 top). Had Trevor moved directly to any position before looking at the display, the same information would have been presented (i.e., he does not have to go through each of the steps above). If Trevor moves away from the thermostat, the process reverses, as a result of gradual disengagement. For fine interaction control, this dynamic updating of information could

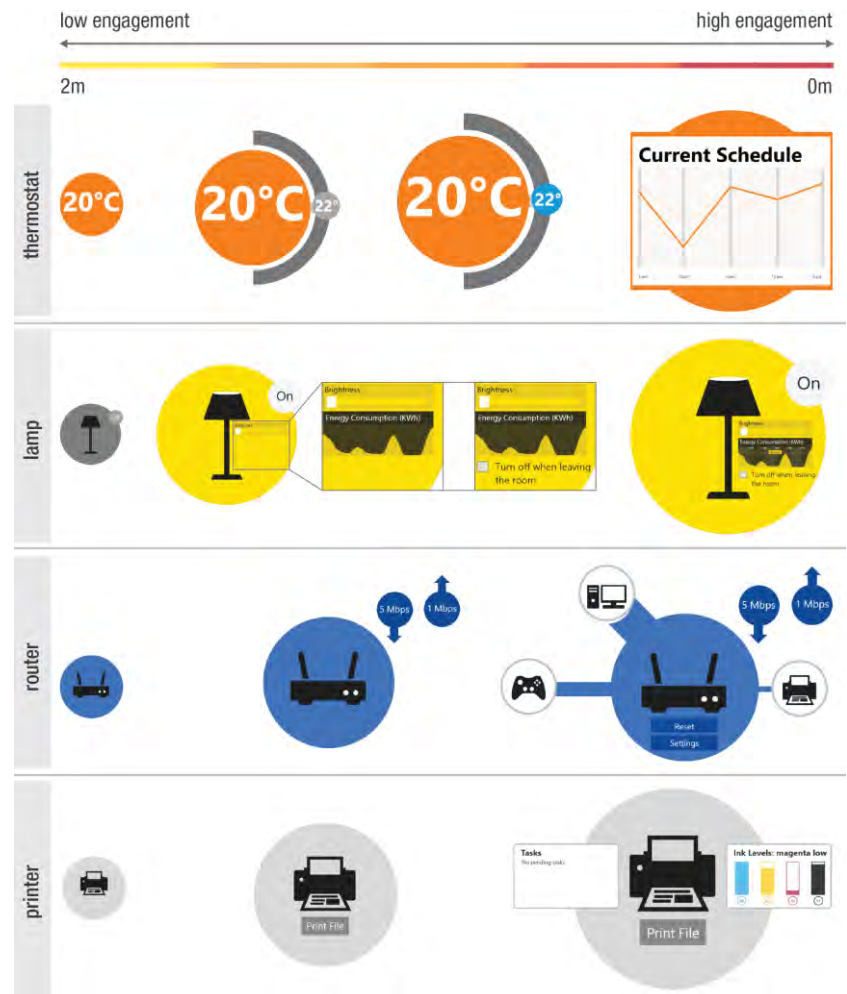


Figure 4. Control interfaces for thermostat, lamp, router and printer at different levels of engagement (distance).

make interaction difficult, so Trevor decided to lock the screen. Locking freezes the interface as it appears at this particular distance and orientation. While not strictly necessary, it allows Trevor to physically move away from the thermostat without changing the interface. While not mentioned in the scenario, Trevor could have switched to another appliance at any time simply by facing towards it.

Scenario 3: Manual Override

Trevor is sitting on his couch watching a movie on the television. He decides to dim his room lighting, but he does not want to get up. He picks up his tablet, and orients it to the lamp which, at that distance, only shows on/off controls (Figure 4, 2nd row left). He locks the interface by pressing the Lock Button, and a 'proximity' slider appears (as in Figure 2, right side). By moving the slider, Trevor manually sets the semantic zoom level as if he had physically moved towards the lamp. He drags the slider until he sees the brightness control, sets it to his desired level, and configures the lamp to turn off when no one is in the room (Figure 4, 2nd row right). Trevor also checks the temperature by manually selecting the thermostat icon on the edge, which makes the thermostat control appear at the center as if he had oriented the tablet towards it.

We mentioned that proxemic-aware controls should complement existing approaches rather than replace them. Unlike the previous scenario, Trevor decided to stay in one place rather than move towards an appliance, as doing so would require extra effort and interrupt his movie viewing. Instead, he locks the interface. Proxemic interactions is disabled, while manual controls allow him to select and control appliances through more conventional means (e.g., the overview icons at the tablet's border (Figure 2) become a graphical menu of selectable appliances, and the Proximity Slider lets him manually navigate the available controls of the selected appliance, revealing progressive detail. Importantly, the appliance interface as revealed by manual override is exactly the same as the proximity-controlled interface.

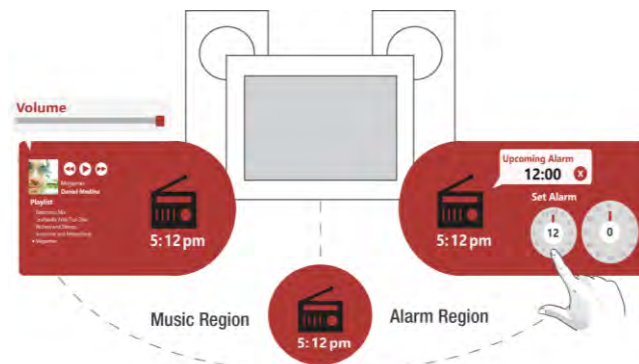


Figure 5. Radio interface at close proximity, showing how different interface details appear when the tablet is oriented at its center and slightly to its left and right.

Scenario 4: Around-Appliance Navigations

Trevor decides to set an alarm before going to bed. He approaches his radio alarm clock, and the tablet shows the radio interface. When he is in close proximity (Figure 5), he shifts his tablet to point slightly to the right of the radio; the interface animates to show a clock control. Using the clock control, he sets the alarm to the desired wake-up time. He then decides to play some music. He shifts the tablet slightly to the radio's left. A more detailed audio interface control appear, and he presses play. Initially, the volume is too low, so Trevor approaches the speakers. This action brings up volume controls which he adjusts accordingly.

Some appliances are quite complex. Thus this scenario illustrates two ways of associating complex information spatially through micro-mobility [16] as yet another way of balancing simplicity and flexibility of controls. The first one is to use *spatial references*, where information connects to a virtual area around the appliance, e.g., controls situated above, below, to the left or to the right. In this example we use left and right to show two different types of controls. However, we note that these spatial references are abstract and must be learned. As a result, they could benefit from feedforward mechanisms. The second type of spatial association is through *semantics*, where specific parts of the appliance signify certain controls. In the radio example, the speakers are inherent to music volume, thus orienting the tablet towards the speakers reveals the volume control (Figure 5).

Scenario 5: Room Viewer Hierarchy

Trevor enters his living room. The entrance of the room acts as a virtual appliance, where the interface shows the room, and the available appliances contained within it (Figure 6); Trevor sees the basic status of each appliance and can adjust a few basic controls for each of them. He selects and turns on the lamp and TV, enters the room, and sits down to watch.

This scenario shows appliances grouped as a hierarchy, where different levels of the hierarchy can be accessed as a

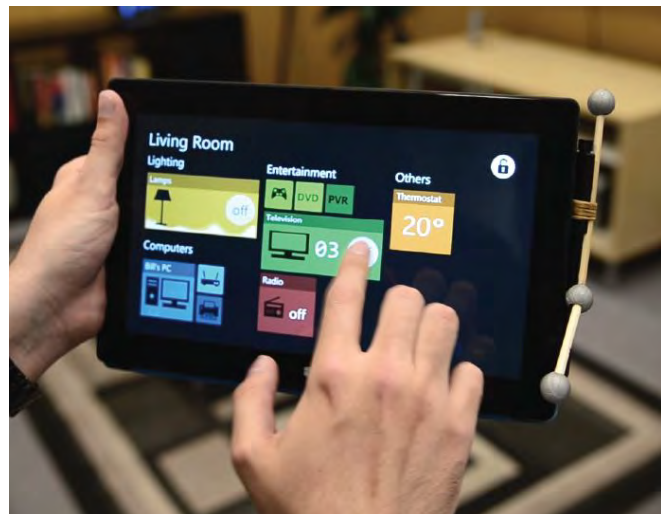


Figure 6. Room Viewer showing all the appliances in the room along with some basic information and controls.

function of proximity. Here, the room entrance serves as a fixed feature [8,1] – a boundary – where the interface displays a high-level at-a-glance view of the contents of the room. The full dynamic interface of Figure 2 would appear only after walking across the boundary. In the Room Viewer, Trevor can see all appliances that are in the room’s ecology and their primary settings. Trevor also has a small degree of control over each appliance, such as being able to switch the television on or off. If he locked the screen on the room view, he is essentially equipped with a control resembling a ‘standard’ universal remote. For example, he can reveal the specific appliance control by tapping on a particular appliance and manually adjusting the Proximity Slider (Scenario 3).

Scenario 6: Situated Context of Actions

The room contains two printers. On the overview, Trevor sees a red exclamation mark next to one of the printer icons, indicating a problem. Since the overview icon spatially indicates the physical printer’s location, he approaches the problematic printer (Figure 4, row 4). A notification appears stating that its ink cartridge is low. After replacing the cartridge, he sees on the tablet that the notification has disappeared, confirming that the printer is now working properly. He decides to print a file to it. While standing next to that printer, a “Print File” dialog appears. He selects a file, which is automatically sent to that nearby printer.

Proxemics spatially situate interaction to their corresponding physical devices, and thus also show notifications in context. We saw an appliance communicate its state by a notification: from afar by flashing an exclamation mark on the overview icon, and on approach where more detail about the notification is progressively revealed. We also saw how proxemics can help disambiguate which appliance of the same type produced the notification. The next part of the scenario demonstrated how the destination of a person’s action can be selected simply by standing next to the desired appliance. In this case, the usual print dialog asking the user to select a printer is not required, as Trevor implicitly selected the desired printer by approaching it. All he needs to do is select the file to print.

Scenario 7: Identity-based Access Levels

Tina, a guest in Trevor’s house, wants to increase the temperature setting of the thermostat. However, while she can see the current temperature and thermostat setting on the remote, the interface to change the setting is not revealed. The reason for this is that Trevor—who is conscientious about reducing his energy use—has configured the thermostat so that only he is able to change its state.

Proxemic-aware controls can leverage an individual’s identity to restrict controls, similar to parental controls but without requiring a password entry. This adds a layer of security to our system. The scenario shows how an unauthorized guest is restricted from controlling the thermostat. Of course, other less restrictive rules can be established, such as allowing Tina (the guest) to change the temperature only if Trevor (the home owner) is co-present. Such an arrangement builds

upon traditional social conventions of people using their own interactions to mediate what the other can do.

RELATED WORK – INTERACTION WITH ECOLOGIES

There has been considerable work in home automation, smart environments, and within the Internet of Things that have also addressed how people can interact with multiple appliances. Our work on proxemics is meant to complement rather than challenge this prior work.

In particular, we have shown a series of scenarios that demonstrate different concepts pertaining to the design of a universal remote control, where emphasis is placed on leveraging the known spatial relationship between the control (the mobile device) and its surrounding appliances. This idea extends previous work highlighting *physical browsing*, usually implemented on mobile devices as a means for people to discover interactive devices and retrieve their corresponding user interfaces [28]. Four of the dominant interaction styles for physical browsing are described below, all which help people associate digital contents to objects in the physical world. Indeed, there are now a broad variety of commercial devices (several of the many available are mentioned below) that use a mix of these physical browsing methods to implement a mobile or surface-based remote control interface to a ‘smart environment’.

Touching

Touching is one known way to associate two devices. The premise is that touching two objects to associate them is easily understood and usually easy to perform by people. Rukzio et al. argue that it reduces accidental selections, and that it is a technique of choice when people are standing, as people prefer to physically approach objects [22]. RFID tags are a common way to implement touching [28,29], though one may also consider synchronous gestures such as bumping two devices that are equipped with accelerometers [9]. Despite the ease of selection, knowing which devices are connectable can be problematic unless they are visibly marked, and thus there is no easy way to preview the scene to see what objects can be associated to each other in the ecology.

Pointing

Pointing a mobile device towards an intended object is appropriate when the two are distant from each other. This technique is enabled by many technologies, such as infrared [4,6,19,26,28], computer vision [11], or light sensing [23]. The advantage of pointing is that the mobile device can display information about the target as soon as it is aligned with it. Other interesting variations exist. For example, InfoPoint enables information from one appliance to be pushed onto another [11]. PICOntrol leverages a mobile projector to reveal an interface with controls overlaid atop of the physical appliance [23]. Chen et al., use a head mounted display to point and reveal context menus for appliances [6], Gestural approaches, such as Charade [2] and Digits [10] focus on arm and hand movement for selection and interaction.

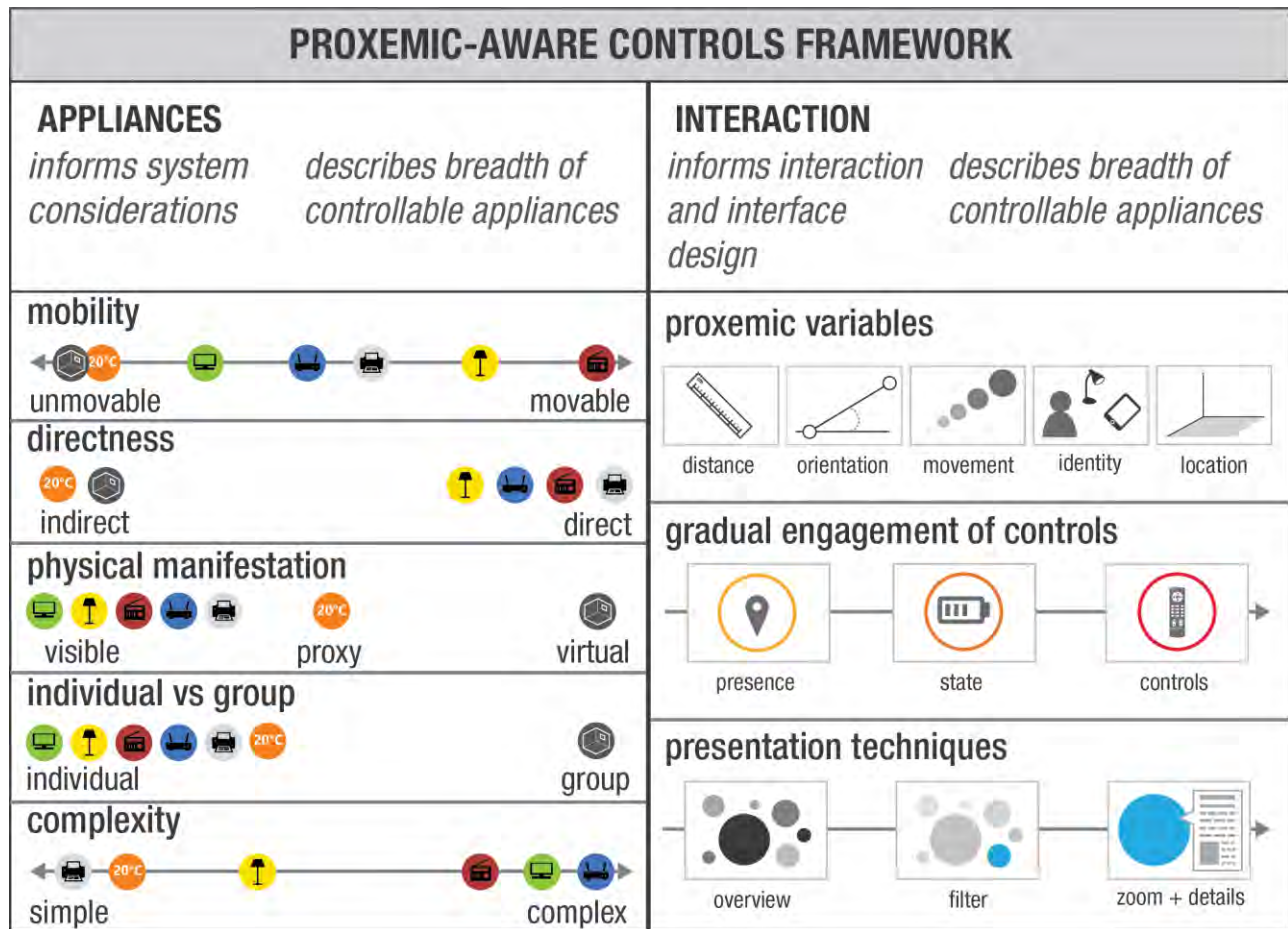


Figure 7. Proxemic-Aware Controls Framework.

Rukzio et al. argue that pointing is a technique of choice when people are sitting [22]. Yet pointing can be problematic with distant targets: small movements can drastically change the pointing direction, thus complicating selection and presenting false-positives.

Scanning

Scanning covers the situation in which a remote control visually displays all appliances it knows about, and then allows the user to select a device to connect or interact with it. Traditionally, scanning makes use of lists [28]. Yet such lists can become difficult to navigate with increasing number of items, as it leads to cognitive overload and difficulty mapping digital content to physical appliances [22], e.g., matching cryptic names to particular appliances. Thus discovery and selection can be difficult.

Scanning is the typical form of interaction seen nowadays with smart appliances, typically through a dedicated mobile app. One example is Nest Thermostat [34], although hubs such as Revolv try to incorporate multiple appliances as a centralized list [33]. Other work, such as Huddle, focuses on using these visual icons to interconnect appliances that operate together [20].

World in Miniature

Another approach is to represent devices through their spatial topography. One way of doing this is through live video feeds in which the interactions with the screen can affect the state of the displayed devices [5,24,27]. For example, CRISTAL present an interactive bird's-eye video view of the room and its controllable devices [24]. Another way to represent topography is through icons showing their relative locations [7,14]. This approach preserves spatial relationships and users thus have an overview of interactive items that facilitates discovery. However, selection can be difficult when presenting a large number of items on a small mobile screen.

Our own method of proxemic-aware controls smoothly combines and extends the above physical browsing methods. Our use of orientation is a method of pointing, and touching is realized as a proxemic distance of 0. The overview at the tablet's edge provides a spatial world in miniature, while moving the tablet around the room to reveal the appliances seen in front of it provides a world in miniature over time. The overview (combined with manual override) allows for scanning, where the list is filtered to show only those appliances in the room.

PROXEMIC-AWARE CONTROLS FRAMEWORK

The scenarios showcased earlier are one instance of a larger design space. Following a ‘research through design’ methodology [31], we transitioned between different design approaches as described by Wiberg and Stolterman [30]. We structured our ideas into concepts and then revealed them as a proof-of-concepts. Our concepts were further abstracted as a conceptual framework called the *proxemic-aware controls framework*. We believe this framework can inform the design of future remote controls. It describes the design space for remote control appliance interaction (discovery, selection, viewing and control) via proxemics as a way to further generalize our investigation (i.e., beyond our own particular implementation), and to place related work in perspective.

The framework describes various dimensions that an appliance may embody (Figure 7, left), and why these may affect how proxemics should be considered. It continues by considering how proxemic theory and visualization techniques can control the interaction flow (Figure 7, right).

Part 1. Appliances

Smart appliance design may vary greatly along several dimensions. As summarized in Figure 7, left, we believe that several dimensions can affect the design of a proxemic-aware controls. Figure 7 left also shows, via representative icons placed on a dimension’s spectrum, how each appliance manifest particular dimensions.

Mobility of an appliance may vary greatly, ranging from unmovable (fixed) to rarely moved (semi-fixed) to highly movable (unfixed). (Hall previously described how such fixed or semi-fixed features can affect interpersonal proxemics [8]). Mobility depends on many factors, including appliance size, weight, wiring (tethered or not), and purpose (which may be location-specific). Examples are a wall-mounted thermostat (unmovable), a router (rarely moved as it is tethered by a cable), a floor lamp (infrequently), and a portable radio or small Bluetooth speaker (moved frequently).

Directness refers to whether a person interacts directly with the appliance, or indirectly through controls representing a perhaps out-of-sight appliance. A typical radio alarm clock is direct, as all input and output controls are found directly on the device. In contrast, a physical thermostat is indirect as it is actually controlling a centralized heating unit located elsewhere. Even so, indirect controls can viewed as a proxy to an otherwise hidden appliance.

Physical manifestation of an appliance affects the user’s ability to visually find and identify the appliance. An appliance is visible when it is physically present in the room, not hidden, and recognizable. If an appliance is indirectly controlled, then it may still be considered visible if its controls are visible (i.e., it serves as a recognizable proxy to the actual appliance). However, smart appliances may also be virtual, where the appliance itself or its controls have no physical manifestation. An example virtual appliance is a sound system comprising speakers embedded into the wall, and that is

only controllable via a dedicated app on a mobile device. Our Room Viewer also acts as a type of virtual appliance, as it virtually groups appliances together into a single appliance.

Individual vs. groups. While most appliances are individual entities, we can also consider an appliance as a set of appliances working together as a group. This was shown in Scenario 5 with the room viewer. Another example is a home theater system comprised of various components, such as a radio, amplifier, television, and media player. Some general / joint actions may apply across the entire group, such as turning them on, and adjusting volume. Other actions will apply to an individual appliance, such as changing a TV’s channel. Remotes such as the Logitech Harmony [35] attempt to combine multiple appliances and show unified controls. Norman refers to this as activity-centered actions, in which the controls are specific to the task a person wishes to perform and which encompasses multiple appliances [21]. Another way to consider grouping is through multiple indirect appliances that perform the same task while being physically scattered, such as ceiling lights in the room. These appliances are often unified through proxies.

Complexity refers to the number of functions that can be controlled and the number of states an appliance can assume. A lamp with only an on/off switch is simple. A more complex version of a lamp would perhaps visualize energy consumption, allow dimming and scheduling, and so on. The radio alarm clock in our system has many controls and states, which makes it an even more complex appliance.

The above dimensions affect the design thinking for proxemic-aware controls. To enable proxemics, the remote control needs to determine distance and orientation to its surrounding appliances. Mobility affects the degree of tracking required for an appliance. For example, we can configure a fixed appliance by setting a one-time location, but a highly mobile appliance may have to be tracked continuously. Directness and physical manifestation implies ambiguities as to what is considered an appliance, and where it is located. This emphasizes the need for thoughtful anchoring of digital information so that people can recognize and spatially associate the location of a virtual appliances to what they see on the screen. For example, we spatially located the Room Viewer virtual appliance at the room’s entrance to provide people with a sense of the interactive appliances contained within the room. Similarly, for grouped appliances, it may be sensible to show universal controls affecting the entire group at a distance, and control individual components as one approaches them. Higher complexity requires thought of how to navigate and progressively reveal an appliance’s controls.

Part 2. Interaction: Proxemics for Remote Control

As described in our design rationale, proxemic interaction serves to situate interaction, provide flexible control, allow for seamless transition between controls, and complement existing types of interactions. Unlike prior explorations of

proxemics in HCI, mobile devices and appliances are a constrained subset of a ubicomp ecology and thus require further contextualization. Figure 7 right summarizes these aspects.

Proxemic Variables

Ballendat et al. proposed a set of variables that inform the design of proxemic interaction: distance, orientation, movement, identity and location [1]. These variables (1) serve as building blocks for interaction, and (2) aid a system's interpretation of people's intents for interaction within the ecology of devices. Our own contextualization of Ballendat et al.'s proxemic variables are described below.

Distance determines the level of engagement between an individual's mobile device and an appliance. This mapping can be *discrete* (different distance thresholds trigger different stages of interaction), or *continuous* (content is revealed on the mobile device as a function of distance, as shown in Scenario 2). Distance, outside of the current work in proxemic interaction, has not been typically considered in prior work to show varying content.

Orientation refers to the direction that an entity is facing with respect to another. It serves to determine if (1) the person is engaging with a particular appliance, and (2) which appliance is the current center of attention. This allows the system to discriminate between pertinent control interfaces to present on the device. The role of orientation is best showcased in Scenario 1. Previous work in pointing uses the orientation relationship as a selection vs. scanning mechanism.

Movement is the change of position or orientation over time. In this context, movement is used implicitly, and thus depends on how fast a user moves their mobile device. Movement incorporates the directionality of the engagement (engaging or disengaging).

Identity uniquely describes the different entities in the space: the people, mobile devices and appliances. The identity of the person can influence the types of control and information presented, such as advanced controls for only the room's owner (as in scenario 7). Mobile devices are tracked continuously and understand their physical relationship with the ecology. Appliances are the target devices for the user, where different users may see different appliance information and capabilities on the user's mobile device.

Location reflects the qualitative aspects of the space that define the rules of social context and behavior. For example, location may influence identity, such as determining groups of appliances (e.g., all those in the room, but none on the other side of the wall), and which persons can control those appliances (e.g., only a person in the room). The physical constraints of the space can also affect the relative measure of proxemic distances and how gradual engagement behaves.

Gradual Engagement of Controls

Gradual engagement is a design pattern describing engagement between a person and a device as a function of proximity [14]. More digital content is displayed on a user's mobile

device as they move closer to an appliance. Our own work focuses on continuous engagement, where interface details of an appliance control are animated to appear or disappear as a function of distance (Figure 4). As described below, we extend and apply gradual engagement as an interaction paradigm to explain people's interaction with discovery, selection, viewing and control of the ecology as illustrated in Figure 4. We also use it to ensure seamless transitions between different appliance interfaces.

Engagement occurs when a person faces and moves toward a target. As the person approaches the target they wish to interact with, they see more related content on their mobile device, which can take the form of information or controls, depending on the appliance and interface design.

Disengagement takes place when a person moves away from a target or appliance. This happens when: (1) the person is moving away from the target while still oriented towards it, thus reversing the gradual engagement; and (2) when an individual is engaged with the target appliance and faces away from it, hence shifting the focus of interaction.

Manual Override or Locking is available when gradual engagement would otherwise be restricting. A shift of focus may happen accidentally if the user's center of attention changes due to small movements on a mobile device (e.g. for more comfortable holding), or for users who wish to remain stationary (e.g., seated) and still be able to control an appliance, as in Scenario 3. Manual override means that users are able to manually (1) pause the current spatial interactions, (2) change the level of engagement, and (3) select any appliance from the ecology and engage with it. This relaxation also integrates *scanning* through manual selection of an individual appliance from an overview; *touching* by approaching a digital appliance to retrieve content; and *pointing* by focusing on an individual appliance through device orientation and manually locking it.

Shifting focus of attention occurs when a person moves their attention from one appliance to another. For example, if a person is viewing an appliance at a certain level of engagement but then re-orient their device to another appliance, that appliance's control appears at the appropriate level.

The next question is how we can apply gradual engagement to content, i.e., what appears within the remote control interface at particular distances. We organized the digital content of an appliance into three categories along the gradual engagement spectrum: *presence (awareness)*, *state (information reveal)* and *controls (interaction opportunities)*. However, we recognize that the interface should impose sharp boundaries between these categorizations, as the interface may present these multiple categories simultaneously.

Presence information refers to the basic identifying information of an appliance. At a high level, an appliance can be thought of as having some sort of label and a location, but this can be further extended by finer-grained descriptions,

such as identifying names, a globally unique identifier, a visual icon that represents the appliance, manufacturer, and type of appliance.

State refers to information describing the current status or behaviour of the appliance. This can be the result of previous actions and controls, or simply the result of current sensor readings, such as a thermostat showing the current temperature of the room. Some state information is immutable, and cannot be changed through controls (e.g., battery levels). State information can go beyond showing the current state. It can show history, such as revealing energy consumption over time, or displaying past actions performed on the appliance. A remote control needs to be capable of displaying such states to provide awareness to the end user.

Controls are appliance states that are changeable. These controls have varying levels of complexity depending on the functionality. A very simple control switches an appliance on or off, while more fine-grained controls allow for discrete values (e.g. light dimmer). More complex controls enable higher customization through settings (e.g. scheduling). Some of these settings can be saved (e.g. favorite channels on a television). Other controls may require information transfer (e.g. printing a file).

The three types of content provide structure and hierarchy. There cannot be state information if the system has no knowledge of the device that the user interacting with (*presence*). Showing *state* information can facilitate *controls* as users can transition from seeing a state to being able to modify it. As a result, the information should build up and increase in complexity as the user gradually engages with an appliance. By making interfaces build up over time, one can ensure a smooth transition from a simple interface to a more intricate and flexible one, thus relaxing the usability versus flexibility trade-off [13].

Presentation Techniques

Unlike traditional user interfaces, proxemics takes *spatiality* into consideration. This means that user interfaces should be dynamic, where it continually reacts as one moves around space. We built upon Ben Shneiderman's mantra of "overview first, zoom and filter, then details on demand" [25] to reveal content and preserving context as a function of gradual engagement. That is, people have to be able to *discover* interactive appliances (overview), *select* one among the ecology (filter), and then *view information and controls* (zoom and details on demand).

Overview corresponds to providing awareness of the interactive appliances present and their relative positions. Spatial references enable discovery. Previous work in ubicomp has mostly presented spatial reference overviews as a bird's-eye view [7,14], a method we used in our own overview which we visually located at the screen's edges. Somewhat similarly, augmented reality research has examined ways to represent other off-screen physical objects in space. Lehtikoinen et al. [12], for example, uses a linear panoramic visualization

to show off-screen targets: the closer they are to the center, the more they are aligned with the center of the field of view. There are, of course, other means of providing overviews (e.g. maps with absolute positioning, scene shrinking [17]).

Filtering takes place by leveraging the user's orientation, i.e. the focus is on the objects that the user is facing. When the user changes their orientation, the position of the appliances and the appliance selected will change accordingly. In our implemented design, we allow for only one appliance at a time (the one in front of the mobile device), where its controls are revealed by animating it to the screen's center.

Zoom and Details on Demand. The distance or proximity between the person and the appliance is a metric that can be used as a mechanism to reveal more content, via a semantic zoom [3]: the amount and detail of content available to the user increases as the distance to the appliance decreases. Similarly, as one approaches a particular appliance, the interface changes dynamically and provides more detailed content. This allows content flow from simplified to complex. However, it can still be difficult to present a large array of controls in close proximity to an appliance because of the mobile device's screen size. This can be addressed with *micro-mobility* [16] (demonstrated in Scenario 4), where some of an appliance's controls are distributed in the space around the appliance to reduce screen navigations and menus.

Referring back to Figure 7, the conceptual framework for proxemic-aware controls structures the variety of appliances that can be controlled (7, left). It also explains how proxemic interaction can be applied to the design of remote controls in a ubicomp ecology (7, right). Gradual engagement of controls frames the interaction flow between a person and an appliance, with the mobile devices acting as interface between the two. Finally, our application of presentation techniques from traditional user interfaces operationalize how gradual engagement occurs within the mobile device.

DISCUSSION

This paper has focused almost exclusively on the idea of proxemic-aware remote controls, and the development of a design space around that idea. While the broad notion of proxemic interactions is not new, we offered various novel ideas within the particular context of proxemic-aware remote controls (e.g., gradual engagement applied to appliance control, micromobility, locking to override proxemics, grouping of appliances into a virtual appliance, etc.). These contributions are best seen as a starting point, where we present a structured approach to a particular design idea.

Reviewers of the submitted version of this paper raised various interesting points as part of their discussion. We include and discuss some of those points here (in paraphrased form), as we believe they are worth airing.

Scalability. Our scenarios are populated with a modest number of appliance (around 5 or 6). Yet it is likely that future ubicomp ecologies could have many more appliances than

that, especially if we come to an era where all electrical devices are digitally accessible. Just consider the number of electrical outlets in a room (all which can have several devices connected to it), the room-oriented devices it may contain (lights, smoke detectors, heating), its specialized appliances (e.g., kitchen vs. home office), the various portable devices that can be brought into that room, and so on. Will spatial organization via proximity scale up to these numbers? We suspect that, at some point, the interface will have to be modified to handle these larger numbers. For example, our idea of proxemics uses orientation (a form of pointing) to target a region of the room. If that region contains a cluster of appliances, the interface would have to handle that. Perhaps the cluster may be portrayed on the interface as a collection, along with a way for the person to select an appliance within that cluster from a distance. While this may mix proxemics with one of the physical browsing mechanisms mentioned earlier, the use of proxemics will still reduce the selection choice to only a subset of appliances within the room.

The metaphor of spatial organization and proxemics. One reviewer wrote: “the point of a remote control is to control appliances from a distance.” Indeed, the tacit idea behind most remote controls is to decouple the control from the appliance (e.g., controlling a television from the couch without having to get up and move towards it). We recognize that spatial decoupling is both powerful and appropriate for various situations. A person may want to operate the appliance from anywhere (e.g., outside the room), or may not even know where the appliance is. A person may also want to manage a group of appliances (e.g., a bank of lights, various parts of an entertainment center) that may be scattered throughout the ecology, and thus have no well-defined physical location. Discoverability via a physical browsing interface provides a guaranteed means to discover what appliances are present and what is controllable vs. browsing the environment via proxemics. In contrast, our idea of proxemic-aware controls re-introduces this spatial coupling, which may seem to some like a step backwards. We recognize that proxemic-aware controls may be appropriate for some appliances and situations (e.g., as discussed in our various scenarios), but not all of them. This is why we see proxemics as complementing rather than replacing with other physical browsing methods, as elaborated next.

Proxemics vs Locking. Reviewers had widely different views about our inclusion of locking. Some thought that locking was an admission that proxemic controls were limited. As one reviewer wrote “Why override everything with sliders? It seems anti-proxemics”. Others had a far more positive view: “the idea of the lock button [is] important and shows they have given thought to the challenges that come with spatially-aware applications and gradual engagement.”. Our stance is that our inclusion of locking recognizes that there is no one ‘pure’ interaction solution to the remote control of appliances. We deliberately included locking – and worked to refine its interface within a proxemic-aware system – because we believed that the two should complement

one another. There are times when physical browsing as operationalized by pure locking is best (which just reverts the system to a traditional remote control), other times when pure proxemics is best, and still others where both can work in tandem (e.g., as described in our scenarios above). Thus we do not advocate proxemics as a ‘one size fits all’ solution. Rather, it is an interaction technique that should complement others.

Tablet vs Phone Form Factor. Another question raised revolved around our use of a tablet as the remote control hardware, rather than a smart phone. While a tablet’s size makes the interface easier to realize and operate, people are much more likely to carry and use a smart phone as the controlling unit. We agree. Indeed, as mentioned earlier in this paper, our first versions were built atop a smart phone. We switched to a tablet largely for pragmatic reasons (e.g., for device tracking, for ease of development, for ease of demonstrating the interface to onlookers, etc.). Even so, we believe that the tablet interface as presented can be largely transferred to a smart phone display (albeit with some modification), with the caveat that very compact smart phones will likely demand an interface redesign to fit its small-screen.

CONCLUSION

This paper introduced proxemic-aware controls as an alternate yet complementary way to interact with increasingly large ecologies of appliances via a mobile device. Through spatial interactions, people are able to discover and select interactive appliances and then progressively view its status and controls as a function of physical proximity. This allows for situated interaction that balances simple and flexible controls, while seamlessly transitioning between different control interfaces. We demonstrated seven scenarios of use, and generalized their broader concepts as a conceptual framework. We believe this a starting point for developing a new type of remote control interface within our increasingly complex ubicomp world.

As with most early work, there is much left to do. While we built a working system, our underlying technology is not suitable for broad deployment. Because our current system is unevaluated (with the exception of observations of casual use and comments made by people trying it out), there are likely design flaws / bugs within our framework and suggested interface that need to be remedied in future iterations. Of course, the real-world viability and acceptance of proxemic-aware controls in practice may depend on many factors that we have not yet addressed.

ACKNOWLEDGEMENTS

This research was funded by AITF, NSERC and SMART Technologies. We thank members of the University of Calgary’s Interactions Lab for their support, and Lora Oehlberg and Jennifer Payne for proof-checking. We also thank the reviewers for their insightful comments, which we partially incorporated into our discussion.

REFERENCES

1. Ballendat, T., Marquardt, N., and Greenberg, S. Proxemic interaction: designing for a proximity and orientation-aware environment. *Proc. ACM ITS 2010*, 121–130.
2. Baudel, T. and Beaudouin-Lafon, M. Charade: remote control of objects using free-hand gestures. *Commun. ACM* 1993, 28–35.
3. Bederson, B.B. and Hollan, J.D. Pad++: A Zooming Graphical Interface for Exploring Alternate Interface Physics. *Proc. ACM UIST 1994*, 17–26.
4. Beigl, M. Point & Click-Interaction in Smart Environments. *Springer HUC 1999*, 311–313.
5. Boring, S., Baur, D., Butz, A., Gustafson, S., and Baudisch, P. Touch projector: mobile interaction through video. *Proc. ACM CHI 2010*, 2287–2296.
6. Chen, Y.-H., Zhang, B., Tuna, C., Li, Y., Lee, E.A., and Hartmann, B. A Context Menu for the Real World: Controlling Physical Appliances Through Head-Worn Infrared Targeting. Report UCB/EECS-2013-200, EECS Department, University of Berkeley, (2013).
7. Gellersen, H., Fischer, C., Guinard, D., et al. Supporting device discovery and spontaneous interaction with spatial references. *PUC, 2009*, 255–264.
8. Hall, E.T. *The Hidden Dimension*. Anchor Books New York, 1969.
9. Hinckley, K. Synchronous Gestures for Multiple Persons and Computers. *Proc. ACM UIST 2003*, 149–158.
10. Kim, D., Hilliges, O., Izadi, S., et al. Digits: freehand 3D interactions anywhere using a wrist-worn gloveless sensor. *Proc. ACM UIST 2012*, 167–176.
11. Kohtake, N., Rekimoto, J., and Anzai, Y. InfoPoint: A Device that Provides a Uniform User Interface to Allow Appliances to Work Together over a Network. *Personal and Ubiquitous Computing* 5, 4 (2001), 264–274.
12. Lehtikoinen, J. and Suomela, R. Accessing Context in Wearable Computers. *PUC 2002*, 64–74.
13. Lidwell, W., Holden, K., and Butler, J. *Universal Principles of Design*. Rockport, 2003.
14. Marquardt, N., Ballendat, T., Boring, S., Greenberg, S., and Hinckley, K. Gradual Engagement between Digital Devices as a Function of Proximity: From Awareness to Progressive Reveal to Information Transfer. *Proc. ACM ITS 2012*, 31–40.
15. Marquardt, N., Diaz-Marino, R., Boring, S., and Greenberg, S. The proximity toolkit: prototyping proxemic interactions in ubiquitous computing ecologies. *Proc. ACM UIST 2011*, 315–326.
16. Marquardt, N., Hinckley, K., and Greenberg, S. Cross-device Interaction via Micro-mobility and F-formations. *Proc. ACM UIST 2012*, 13–22.
17. Mulloni, A., Dünser, A., and Schmalstieg, D. Zooming interfaces for augmented reality browsers. *Proc. ACM MobileHCI 2010*, 161–170.
18. Myers, B. Mobile Devices for Control. *Springer Human Computer Interaction with Mobile Devices*, 2002.
19. Myers, B.A., Peck, C.H., Nichols, J., Kong, D., and Miller, R. Interacting at a Distance Using Semantic Snarfing. *Proc. Springer Ubicomp 2001*, 305–314.
20. Nichols, J., Rothrock, B., Chau, D.H., and Myers, B.A. Huddle: automatically generating interfaces for systems of multiple connected appliances. *Proc ACM UIST 2006*, 279–288.
21. Norman, D.A. *Living with Complexity*. MIT Press, 2010.
22. Rukzio, E., Leichtenstern, K., Callaghan, V., Holleis, P., Schmidt, A., and Chin, J. An Experimental Comparison of Physical Mobile Interaction Techniques: Touching, Pointing and Scanning. *Proc. Springer Ubicomp 2006*, 87–104.
23. Schmidt, D., Molyneaux, D., and Cao, X. PIControl: using a handheld projector for direct control of physical devices through visible light. *Proc. ACM UIST 2012*, 379–388.
24. Seifried, T., Haller, M., Scott, S.D., et al. CRISTAL: a collaborative home media and device controller based on a multi-touch display. *Proc ACM ITS 2009*, 33–40.
25. Shneiderman, B. The eyes have it: a task by data type taxonomy for information visualizations. *Proc. IEEE VL/HCC, 1996*, 336–343.
26. Swindells, C., Inkpen, K.M., Dill, J.C., and Tory, M. That One There! Pointing to Establish Device Identity. *Proc. ACM UIST 2002*, 151–160.
27. Tani, M., Yamaashi, K., Tanikoshi, K., Futakawa, M., and Tanifuji, S. Object-oriented video: interaction with real-world objects through live video. *Proc. ACM CHI 1992*, 593–598.
28. Välikynen, P. and Tuomisto, T. Physical Browsing Research. *Proc. PERMID 2005*, (2005), 35–38.
29. Want, R., Fishkin, K.P., Gujar, A., and Harrison, B.L. Bridging physical and virtual worlds with electronic tags. *Proc. ACM CHI 1999*, 370–377.
30. Wiberg, M. and Stolterman, E. What Makes a Prototype Novel?: A Knowledge Contribution Concern for Interaction Design Research. *Proc. ACM NordiCHI 2014*, 531–540.
31. Zimmerman, J., Forlizzi, J., and Evenson, S. Research Through Design As a Method for Interaction Design Research in HCI. *Proc. ACM CHI 2007*, 493–502.
32. Remote Control Anarchy (Jakob Nielsen’s Alertbox). 2004. <http://www.useit.com/alertbox/20040607.html>.
33. Revolv. Accessed October 2014. <http://revolv.com/>.
34. Nest. Accessed February 2015. <https://nest.com>.
35. Logitech Harmony. Accessed February 2015. <http://www.logitech.com/en-ca/product/harmony-ultimate-home>.