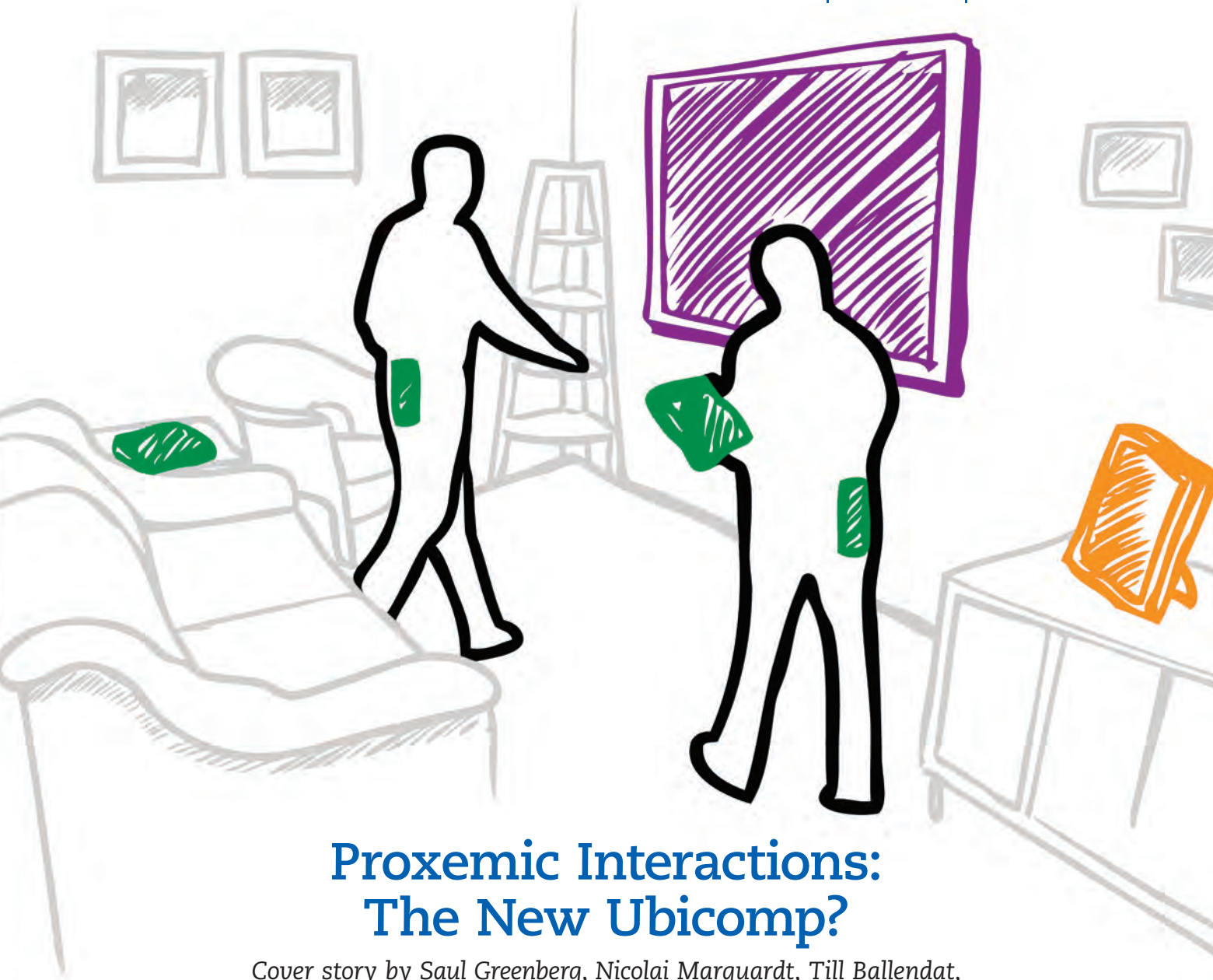# interactions

Proxemic Interactions:
The New Ubicomp?

Cover story by Saul Greenberg, Nicolai Marquardt, Till Ballendat,
Rob Diaz-Marino, and Miaosen Wang

▶ PAGE 42

Proxemic Interactions: The New Ubicomp?

Saul Greenberg
University of Calgary | saul.greenberg @ucalgary.ca

Nicolai Marquardt
University of Calgary | nicolai.marquardt@ucalgary.ca

Till Ballendat
University of Calgary and Ludwig-Maximilians-Universität | till@ballendat.info

Rob Diaz-Marino
University of Calgary| rob.diazmarino@gmail.com

Miaosen Wang
University of Calgary | wangmiaosen@gmail.com

"When you walk up to your computer, does the screen saver stop and the working windows reveal themselves? Does it even know if you are there? How hard would it be to change this? Is it not ironic that, in this regard, a motion-sensing light switch is "smarter" than any of the switches in the computer…?
—Bill Buxton [1]

In 1966 anthropologist Edward Hall coined the term "proxemics," an area of study that identifies the culturally dependent ways in which people use interpersonal distance to understand and mediate their interactions with other people [2]. While his theory of proxemics has many aspects to it, perhaps the most relevant to HCI are his definitions of four proxemic "zones," which characterize how people interpret interpersonal distance: intimate (less than 1.5 feet), personal (1.5 to 4 feet), social (4 to 12 feet), and public (12 to 25 feet). As these

names imply, closer distances lead to increasing expectations of interpersonal engagement and intimacy. In practice, people adjust these distances not only to match their social activities, but also to raise defense mechanisms when others intrude into these zones. Hall also described how features within the space affect people's interactions. "Fixed features" include those that mark boundaries (e.g., entrances to a particular type of room), where people tend to organize certain kinds of social activities within these boundaries. "Semi-fixed features" are entities whose position can affect whether the space tends to bring people together or move them apart (for example, the arrangement of chairs).
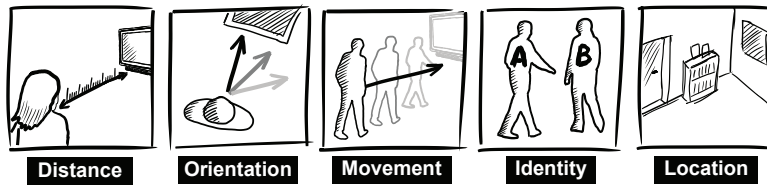
To understand why this is important to ubiquitous computing (ubicomp), we need to revisit the ubicomp vision. In 1991 Mark Weiser—recognized as the founder

Digital surfaces

Portable personal devices

People

Information appliances

► Figure 1: A proxemic ecology, including a mix of people, digital surfaces, portable personal devices, and information appliances.

▶ Figure 2; The five dimensions of proxemics for ubicomp.

Distance **Orientation** **Movement** **Identity** **Location**

of ubicomp—described it as technologies that disappear, that "weave themselves into the fabric of everyday life until they are indistinguishable from it," where computers are integrated "seamlessly into the world" [3]. He envisioned many computers per person, all interconnected. The form factor of the device would heavily influence what it would be used for: inch-scale displays as notes, foot-scale displays as paper, yard-size displays as whiteboards. Devices would know about their location and surrounding, where behavior and function would depend to some extent on environmental context (we now call this context-aware computing).

Twenty years later, it appears that we have arrived at Weiser's vision, what with the common use of smartphones, tablets, laptops, large digital touch surfaces, and other information appliances. Yet we haven't. There are still considerable problems that make these devices far from seamless. For example, consider the digital ecology of the living room shown in Figure 1. It includes various devices (the digital surface, the information appliances, and the things people carry, such as smartphones and tablets). While most devices are networked, actually interconnecting these devices is painful without extensive knowledge, and it requires time to configure and debug. Even when devices are connected, performing tasks among them is usually tedious—for example, navigating through network and local folders

to find and exchange files. In practice, this means that, from a person's perspective, the vast majority of devices are blind to the presence of other devices. What makes this even more problematic is that these devices are also blind to the non-computational aspects of the room—the people, other non-digital objects, the room's semi-fixed and fixed features—all of which may affect their intended use. While a portable device may recognize that another device is in range (e.g., via Bluetooth), it cannot tell if that second device is in the same room or a different one.

This is where proxemics can help. Just as people expect increasing engagement and intimacy as they approach others, so should they naturally expect increasing connectivity and interaction possibilities as they bring their devices in close proximity to one another and to other things in the ecology.

Operationalizing Proximity for Ubicomp
Before jumping into things, we need to operationalize the concept of proximity in ubicomp, that is, to make proximity measurable. Hall's theory of proxemics saw interpersonal distance encompassing not only pure distance, but social and cultural elements as well. Ubicomp proxemics is somewhat different, as it concerns inter-entity distance, where entities can be a mix of people, digital devices, and non-digital things. Since we want to design ubicomp applications that

somehow sense proximity, we have to be clear about what measures proximity will include. Our own notion of proxemic dimensions for ubicomp is characterized in Figure 2 and explained below. Each of these dimensions can also be considered in a variety of ways, suggesting measures that can vary by fidelity and the values they return—discrete or continuous.

• *Distance* between entities is fundamental. We normally think of distance as a continuous measure, such as a value returned between zero and six feet. However, distance can also be discrete. As with Hall's proxemic zones, others have defined specific zones between devices along with implications of what the zone means. For example, Vogel and Balakrishnan defined four "interaction" zones that affect how a digital vertical surface should react to one or more approaching people [4], e.g., supporting ambient display in the outermost zone and supporting explicit personal interaction in the innermost zone (see Figure 3). In these cases, "distance" is a discrete measure of what zone an entity is in with respect to another entity. In the simplest case, this is just a binary measure, e.g., one entity is or is not in the same room as another entity.

• *Orientation* between entities captures nuances not provided by distance alone. It too can be continuous (e.g., the pitch/roll/yaw angle of one object relative to another), or discrete (e.g., facing toward, somewhat toward, or away from the other object). Orientation as an input measure has already been applied to sensing attention in attentive user interfaces, where a device recognizes and takes action when a person is looking at it [5]. Of course, orientation makes sense only if an entity has a "front face."

• *Identity* uniquely describes the entity. This can range from a detailed measure, including exact identity and attributes, to a less detailed measure, such as an entity's type, to a minimal measure that simply distinguishes one entity from another.

• *Movement* captures the distance and orientation of an entity over time, where different actions can be taken depending on, for example, the speed of motion and whether one entity is moving and turning toward versus away from another entity.

• *Location* describes the physical context in which the entities reside, for example, a particular room and its characteristics. Location measures can capture contextual aspects, such as when an entity crosses a threshold (a fixed feature), marking its presence in a room. Location is important, as the meaning applied to the four other inter-entity measures may depend on the contextual location.
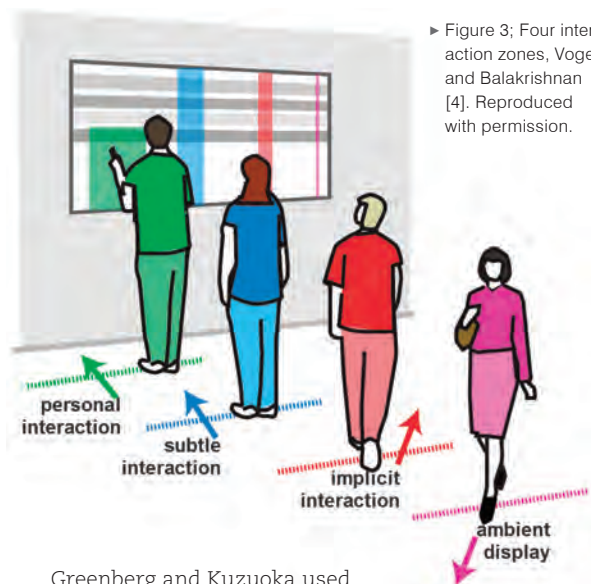
Some of these measures have appeared before in other ubicomp systems, but very few make use of all of them, let alone consider them as characterizing the interplay between entities in an ecology. The idea of a new ubicomp thus arises from our use of these proximity dimensions in system design.

Using these dimensions, we now illustrate how this new, nuanced ubicomp works through several examples of systems built in our laboratory over the years. As we will see, some of our systems use only relative distance, while others add knowledge of orientation, movement, and identity. Most are designed for a particular location. We should also mention that we are not the first to investigate proxemics in ubicomp. Others that have influenced our own work include

Dan Vogel (U. Toronto), Wendy Ju (Stanford), Ken Hinckley (Microsoft Research), Hans Gellersen (Lancaster U.), Peter Tandler and Norbert Streitz (Fraunhofer), Garth Shoemaker (U. British Columbia), Jeremy Cooperstock (McGill), George Fitzmaurice (Autodesk), Jun Rekimoto (U. Tokyo), and many more.

Mediating Video Communication: The Proxemic Media Space

In the late 1990s, Saul Greenberg and Hideaki Kuzuoka experimented with proximity as a way to control an always-on audio/video connection (a.k.a. a media space) between distance-separated colleagues [6]. The motivation behind always-on video/audio is that it becomes a channel that provides awareness of a colleague's presence and activities. In turn, this awareness creates opportunities for colleagues to easily move into casual conversations and interactions over that same channel. Our particular interest was to mitigate privacy and distraction concerns endemic to most media spaces. Specifically, we used proximity as a way for people to naturally adjust the balance between awareness and privacy. We built small audio/video units outfitted with simple sensors that measured a person's distance from the unit (see Figure 4); the audio and video fidelity was controlled as a function of each person's position relative to the device. Mimicking Hall's interpersonal proximity zones, both people could see and hear each other at full fidelity when they were both close to units. As one or both moved away from their units, audio was disabled (lending some privacy), while moving even farther away degraded the video to the point that each knew the other was present but could not see much detail.

Figure 3; Four interaction zones, Vogel and Balakrishnan [4]. Reproduced with permission.

Greenberg and Kuzuoka used these units to communicate between their offices. To make them work, each had to position it within his office in a location that made sense. Greenberg, for example, had it on his desk to the right of where he usually worked. His distance from the display while working allowed an awareness of Kuzuoka's presence over the channel. When he wanted to talk, he just leaned toward the unit, which opened the audio channel and increased the video fidelity, prompting Kuzuoka to respond by leaning toward his own unit, completing the two-way connection. When another

Figure 4: A proxemic-aware media space. Video and audio fidelity are adjusted as a function of distance measured by a range sensor.

▶ Figure 5: The Proxemic Face as a social entity. (a) The lonely proxemic face. (b) It sees Rob come in and greets him. (c) It looks at Rob when Rob looks at him (d) but is saddened when Rob looks away. (e) Initially fascinated by the flashlight beam, it is annoyed when Rob pokes it in the eye. (f) Rob is a bit too close for comfort.

person entered his office, Greenberg would usually move toward a small table away from his desk (and thus the unit), which degraded the video—Kuzuoka knew that a conversation was occurring but could not see or hear any details. While this explanation is technical, in practice both found this a very easy and socially natural way to interact while still maintaining some privacy and minimizing distraction.

The Social Surface: The Proxemic Face

Our next major project on proxemics was created to demonstrate the capabilities of our "Proximity Toolkit" for rapidly prototype proxemic interactions (see sidebar). To test our toolkit, we decided to build a social actor—a caricature—whose behavior was driven by a set of simple rules inspired by Hall's proxemic zones. The sequence below illustrates some of its behaviors

(see Diaz-Marino and Greenberg's "The Proximity Toolkit and ViconFace: The Video" [7]). In Figure 5 we see (a) the proxemic face is lonely when no one is present, (b) happy when its friend comes into the room, (c-d) maintaining eye contact and expression as a function of distance, (d) becoming sadder as its friend moves or looks away, (e) annoyed when its friend pokes it in the eye, and (f) becoming angry as its friend crosses into his intimate space. The face was also startled by sudden movements and could be distracted by other objects pointed toward it. While the face was just a simple social caricature, visitors to our lab found it immediately understandable and compelling, where they assumed it had much more intelligence and knowledge of social rules than it actually had (its behavior repertoire was really nothing more than a simple state machine).

Information and Controls in Hand: Proxemic Presenter

Our next project was more application-oriented. We wanted to see what we could do if we added proximity awareness to a traditional presentation tool (e.g., PowerPoint) running on a vertical surface. We focused on two specific capabilities: We wanted to make it easier for a speaker to access his or her speaker notes, and we wanted to make it easier for a speaker to jump over slides by selecting from a set of overview thumbnails. While existing tools have these capabilities, they usually work best through a second display.

Miaosen Wang created the "Proxemic Presenter" to provide these facilities directly on the single surface. It exploits distance, orientation, and identity (to distinguish the speaker from others). The sequence in Figure 6 shows how it works. (a) When a speaker is facing the audience, the presentation fills

Prototyping Proxemic Interactions: The Proximity Toolkit

There are many ways to capture proximity data. Methods include sensors, vision and scene analysis, motion capture via tags, time-of-flight measures, instrumented rooms, depth sensors, and others. No method is yet perfect, as there is a trade-off between important factors such as data accuracy, the type of information returned, equipment costs, difficulty of configuration, and amount of custom coding required to exploit the returned information effectively.

Because we wanted to concentrate on the design of proxemic interactions instead of the underlying plumbing, we built the Proximity Toolkit. Currently based on the expensive Vicon Motion Capture system, it tracks particular objects (via markers) and their proximity relationships with each other. From that, we generate highly accurate

distance, orientation, identity, and movement information as a series of easy-to-program events. Additional information processed from this data is also returned as events, such as the intersection ray of one object facing toward another object, or whether one object has "collided" with another object by crossing a distance threshold. Programming with these events is straight-forward. We found that computer science students, after just an hour of training, could construct simple but quite interesting proximity-aware applications in a very short amount of time (a day or two).

Figure 13 illustrates one of the controls in this toolkit, where it is displaying the current state of the living room ecology described in previous systems. The figure shows the fixed and semi-fixed features of the room (the room boundaries, the coach, side table, bookcase, and displays). It also dynamically shows the several moving entities in the room and their orientation (a wand and the person by his hat), and that the person is touching the display. Programmatically, it

the screen as expected. (b) When the speaker stands at the side of the screen and turns toward it, a small but readable pane containing speaker notes, timing information, and next/previous controls fades into view next to the speaker. (c) As the speaker looks back toward the audience, the notes pane fades away. (d) The notes pane follows the speaker: If the speaker moves to the other side of the display and looks toward it, the pane appears at that side. (e) If the speaker moves away from the display and then looks toward it, the notes pane does not appear. This is because the speaker is too far away to read them, and showing large notes would be distracting to the audience. (f) If the speaker shields the display from the audience by standing near and at the center of the surface, a scrollable deck of slide thumbnails appears, allowing the speaker to rapidly switch to any slide.

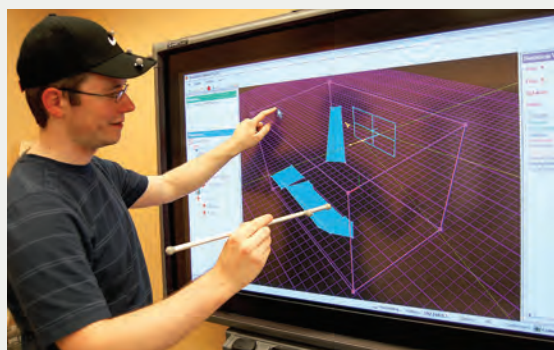**Multiple People and Devices: The Proxemic Media Player**

The above systems considered only the spatial relationships between two entities in an ecology (a person and a surface). Till Ballendat and Nicolai Marquardt's effort was to consider the broader room ecology of multiple people, multiple devices, and even non-digital devices. The test bed was a media player that ran on a large display.

Through a set of scenes described below, we will see how the proxemic media player reacts to the proxemic relationships of two people; non-digital objects, including a cell phone and pencil; digital devices, such as a personal media player; and to the room's fixed and semifixed features, including the entranceway, a couch, and a large digital surface (for details plus video, see Ballendat, Marquardt, and Greenberg's "Proxemic Interaction: Designing

for a Proximity and Orientation-Aware Environment" [8]).

Person to surface. Our scenario follows Till; Figure 7 (top) shows where Till is in a room, and Figure 7 (bottom) shows what the surface is displaying at those distances from the surface. The related surface displays are as follows: (a) Till enters the room. The media player recognizes Till's identity, activates the display, shows a short animation, and then displays four large video preview thumbnails held in Till's personal media collection at a size suitable for distance viewing. (b) As Till moves closer to the display, it shows an increasing number of his videos by continually shrinking the video preview thumbnails and titles to a smaller size. (c) When Till is very close, he can select a video to watch directly by touching its thumbnail on the screen, which shows him more about the selected video: a

▶ Figure 6:
The Proxemic Presenter. (a) While Miaosen presents, (b) he turns to the screen to see his speaking notes; the slide controls fade in next to him, (c) which fade away as he looks back to the audience. (d) He switches sides, looks back to the screen, and the notes appear next to him on that side. (e) When standing farther away and looking toward the display, the notes do not appear. (f) But when Miaosen approaches the middle of the display, a scrollable slide deck appears, and he can skip to particular slides.

continuously provides the relative proxemic dimensions of tracked objects. Specifically, any object can be tracked and identified by attaching a unique arrangement of markers to it. For example, markers on baseball caps uniquely identify their wearers. Markers on a cell phone or tablet uniquely identify that cell phone or tablet. Markers on a wand (a

▶ Figure 13: One view of the Proximity Toolkit in action.

pointer) identify that wand. The toolkit also lets one configure the location of semi-fixed and fixed features in the ecology (stored internally as a 3-D model), where the proximity relationships between any object and those features are also returned. For example, the model may contain the fixed-feature position of the entranceway to a room, allowing one to know if someone has crossed that threshold. It may also contain the location of semi-fixed features, such as the couch and touch-sensitive large digital surface. Unlike objects that move around, these features stay in the same place, and thus, their position does not have to be tracked dynamically.

We predict that accurate proximity information will soon be available to most developers and consumers, particularly through affordable game consoles such as Microsoft's Kinect, which uses a depth camera for its input sensor. Our toolkit anticipates this, where its internal structure is set up to accept sensing information from any source and abstract it to the five key proximity parameters mentioned above.

► Figure 7: How the media player reacts to a person's proximity.

► Figure 8: The media player interprets a non-digital device as a pointer.

preview that can be played and paused, detailed title, authors, description, and release date. The text is small but quite readable at this close distance. (d) When Till moves away from the screen to sit on the couch, his currently selected video expands to play in full screen view. Playback is resumed where Till left off; otherwise it starts from the beginning. While Till's distance from the screen is similar in (a) and (d), the system tells them apart because (a) is associated with a fixed feature of the space (the entrance threshold), while (d) is associated with a semi-fixed feature (the couch).

Non-digital device to surface. Till tires of this video and decides to select a second video from the collection. He pulls out his cell phone and points it toward the surface (see Figure 8). The system recognizes it as a pointer directed at the surface, based upon the phone's distance from the person and its orientation to the surface. The surface shrinks the running video somewhat to show a row of preview videos at its bottom. A visual pointer on the screen provides feedback of the exact pointing position of Till's phone relative to the screen. Till then selects the desired videos by flicking the hand downward, and the video starts playing. Alternately, Till could have used another pointing object (such as a non-digital pen) to do the same interaction.

The surface reacts to people's attention. Figure 9 shows the various ways in which the media player reacts to inattention. (a) Till turns away from the screen to read a magazine. After a few moments, the system interprets this new orientation as a lack of attention and automatically pauses the video. When Till turns to look back at the screen,

playback resumes. (b) Till receives a phone call; he answers. The system recognizes the proximity and orientation of the cell phone to Till as a call and pauses the video. It resumes playback after he finishes the call and puts the phone in his pocket. (c) If Till and another person are facing each other for a while, the system recognizes this as a conversation and also pauses.

Digital device to surface. Figure 10 illustrates, progressing from left to right, what happens as Till pulls out his media player, orients it toward the surface, and approaches it. (a) Till takes his personal portable media player from his pocket. A small graphic representing the mobile device appears on the border of the large display, indicating that media content can be shared between the surface and portable device. The device's position relative to the display is tracked; its graphic moves horizontally across the surface to be as close to the physical device as possible. (b) Till moves closer to the surface while orienting his device toward it; the graphic on the surface responds by progressively and continuously revealing more information about the content held on the media device. (c) When Till moves directly to the front of the surface while holding the device, he sees large preview images of the device's video content and can then transfer videos to and from the surface and portable device by dragging and dropping their preview images. The video playback on the large screen resumes as Till puts his portable device back in his pocket and sits down on the couch.

A second person to the surface. Figure 11 illustrates how the media player adjusts what it displays so that it is appropriate to the proxemic relations of two people to it.

(a) As Till sits on the couch and watches the video, Nic enters the room. The title of the currently playing video shows up at the top of the screen to tell Nic what video is being played. (b) When Nic approaches the display, more detailed information about the current video becomes visible at the side of the screen where he is standing—if he moves to the other side, the description will reappear there. (c) When Nic moves directly in front of the screen (blocking Till's view), the video playback pauses and the browsing screen is shown. Nic can now select other videos by touching the screen. The player changes back into full-screen view once Nic and Till both sit down to watch the video. We already described how if Till and Nic start talking to each other, the video will pause until one of them looks back at the screen. When both leave the room, the application stops the video playback and turns off the display.

Game Play: Proxemic Pong

Atari's Pong, originally created in 1972, is a tennis-based sports game: A person hits a moving ball with a paddle, the ball bounces off the walls, and then the other person tries to hit the returning ball until someone misses. What if this game could exploit proxemics? As a fun side-project, Ballendat created Proxemic Pong, running on a vertical surface (see Figure 12). The game reacts to distance, orientation, motion, and identity, where identity just distinguishes between different players. In standby mode, which displays a splash screen, Proxemic Pong recognizes when a person enters and stands in front of the screen. It creates a paddle for that person and starts the game. The player controls the paddle with their body by facing forward and moving side to side. When a second person stands in front of the display, a second paddle is created and the game continues via turn-taking (as seen in Figure 12). To penalize the player who interferes with the active player by standing in their way, Proxemic Pong enlarges the active player's paddle to make it easier to hit the ball.

Like Wii games, Proxemic Pong introduces an exertion element into computer game play. Initially, the player's motion matches the paddle's motion. As game play continues, the system increases the ratio of the physical distance that needs to be covered to move the paddle, while also increasing the speed of the ball. This means that people have to move farther and faster to hit the ball.

Proxemic Pong also exploits front-to-back motion. If a player moves very close to the display, the game automatically pauses; control points appear on the paddle, allowing that person to adjust the paddle shape by direct touch (see Figure 12, inset). If a player moves backward and sits on the couch (i.e., the player becomes an observer), his or her paddle disappears and the game continues in single-player mode. If both move away, the game pauses.

The New Ubicomp?

In this age of touting natural user interfaces, proxemic interactions has great potential. If well designed, it can exploit people's expectations of how they and their devices should interact within particular ecologies as they move toward one another. But there is still much left to do, and thus many uncertainties.
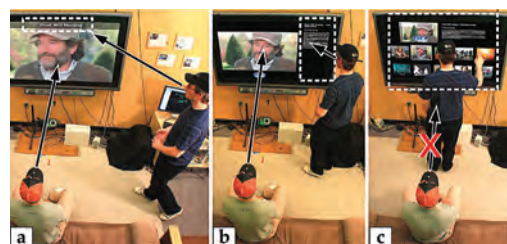
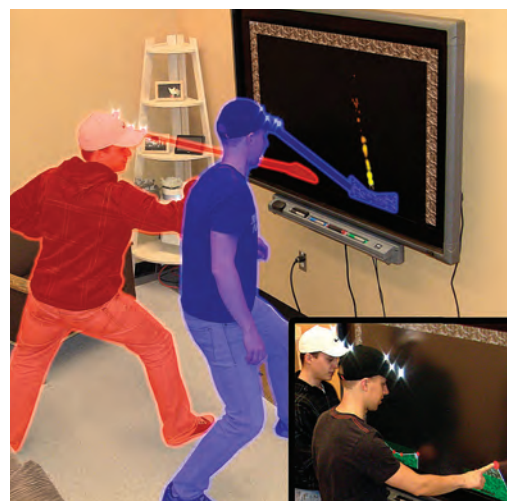First, many of the interaction techniques revolve around how an

▶ Figure 9: The media player interprets various proxemic relationships as inattention.

▶ Figure 10: The media player displays increasing information and opportunities for interaction as the media player approaches and is oriented toward the surface.

▶ Figure 11: The media player changes what it displays to be appropriate to the proximity of two viewers.

▶ Figure 12: Proxemic Pong in action. The inset shows how people can adjust paddle shape when close to the screen.

entity acts, based on its interpretations of the acts of another entity. That is, it assumes that a set of rules of behavior exists to dictate what that entity should do based on implicit acts versus some states in which a person controls interaction directly through explicit acts. While it is easy to create believable scenarios where a rule set makes sense, there will always be many cases where applying the rule in a particular instance will be the wrong thing to do. This raises the question of how one goes about designing (or dynamically learning, via an AI and/or machine-learning approach) these rules of behavior. It also raises the question of how a person could control such systems; indeed, this implicit/explicit interaction control was the primary concern of Ju, Lee, and Clemmer [9].

Second, sensing systems are just guesses into what is actually going on in the environment. Most inputs (at least in the near future) will likely be low fidelity, limited, contain many inaccuracies (including noise), will miss critical information, and so on. Designing robust proxemic interactions around inaccurate or incomplete proximity information will be challenging.

Third, while a growing number of people are investigating how proximity can be applied to interaction design, this is still fairly new work. We just don't understand the HCI of proxemics. Hall's theory is at best suggestive to design. While a theory of proxemic interaction is intuitively appealing, creating one that describes and explains people's expectations of ubicomp is work for the future.

In spite of these misgivings, we can create simple and effective proxemic interaction systems today. Coming full circle from the

Buxton quote that introduced this article, Miaosen Wang used clay to attach a very cheap Phidget range sensor [10] to the side of a computer display. Via a short computer program that display now turns itself off if no one is sitting in front of it and turns itself on again when someone returns (available from http://grouplab.cpsc.ucalgary.ca/cookbook/index.php/Demos/Proximity-MonitorEnergy/). This is affordable stuff that could be built into every screen. While the technology is crude, the savings in terms of cost and the environment could be enormous. As Buxton suggested, there is potentially great merit in making our computers at least as smart as light switches and toilets.

Acknowledgments

ENDNOTES

[1] Buxton, W. Living in augmented reality: Ubiquitous media and reactive environments. V*ideo Mediated Communication.* K. Finn, A. Sellen, and S. Wilber (eds.). Erlbaum, Hillsdale, N.J, 1997.

[2] Hall, E.T. *The Hidden Dimension.* Doubleday, New York,1966.

[3] Weiser, M. The computer for the 21st Century. *Scientific American* 265 (1991), 94-102.

[4] Vogel, D. and Balakrishnan, R. Interactive public Aambient displays: transitioning from implicit to explicit, public to personal, interaction with multiple users. *Proc. of the 17th Annual ACM Symposium on User Interface Software and Technology.* (Santa Fe, NM, Oct. 24-27). ACM, New York, 2004,137-146.

[5] Shell, J.S., Selker, T., and Vertegaal, Interacting with groups of computers. *Comm. ACM 46*, 3 (Mar. 2003),40-46

[6] Greenberg, S. and Kuzuoka, H. Using digital but physical surrogates to mediate awareness, communication and privacy in media spaces. *Personal and Ubiquitous Computing 3*, 4 (1999), 182-198.

[7] Diaz-Marino, R. and Greenberg, S. The Proximity Toolkit and ViconFace: The Video. *Proc. of the 28th of the International Conference Extended Abstracts on Human Factors in Computing Systems.* (Atlanta, GA, Apr. 10-15). ACM, New York, 2010.

[8] Ballendat, T., Marquardt, N. and Greenberg, S. Proxemic interaction: Designing for a proximity and orientation-aware environment. *Proc. of the ACM International Conference on Interactive Tabletops*

and Surfaces 2010. (Saarbruecken, Germany, Nov. 7-10). ACM, New York, 2010.

[9] Ju, W., Lee, B.A., and Klemmer, S.R. Range: exploring implicit interaction through electronic whiteboard design. *Proc. of the 2008 ACM Conference on Computer Supported Cooperative Work* (San Diego, CA, Nov 8-12). ACM, New York, 2008, 17-26.

[10] Greenberg, S. and Fitchett, C. Phidgets: Easy development of physical interfaces through physical widgets. *Proc. of the 14th annual ACM Symposium on User Interface Software and Technology.* (Orlando, FL, Nov. 11-14). ACM, New York, 2001, 209-218, To appear in *Proc. of MobileHCI 2010.*

ABOUT THE AUTHORS

Saul Greenberg is a professor in the Department of Computer Science at the University of Calgary. He holds the NSERC/iCORE/Smart Technologies Industrial Chair in Interactive Technologies and a University Professorship—a distinguished University of Calgary award recognizing research excellence. He received the CHCCS Achievement Award in May 2007 and was also elected to the prestigious ACM CHI Academy in April 2005 for his overall contributions to the field of human-computer interaction. http://www.cpsc.ucalgary.ca/~saul/

Nicolai Marquardt is a Ph.D. student at the University of Calgary and a former intern (twice!) at Microsoft Research, Cambridge. His dissertation directly focuses on proxemic interactions within a ubicomp ecology. http://www.nicolaimarquardt.com/

Till Ballendaat is a diploma student in media informatics at LMU (Ludwig-Maximilians-Universität), Munich (Germany). His proximity work was performed during a six-month research internship at the University of Calgary. http://www.tillballendat.de/

Roberto Diaz-Marino was a research assistant working with Saul Greenberg at the University of Calgary. He has recently moved on to SMART Technologies, Inc.

Miaosen Wang is an M.Sc. student supervised by Saul Greenberg, with prior experiences working at SMART Technologies, Inc., on interaction techniques for surfaces. His thesis topic concerns how a surface can work as an attractant leading to interaction as people move toward the surface.