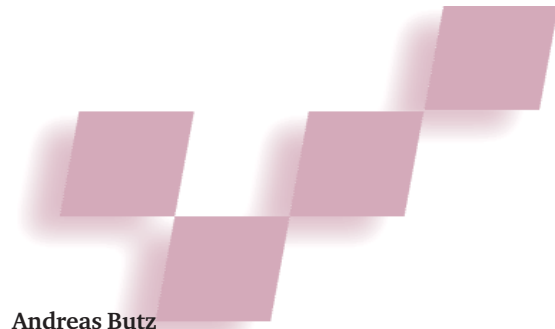


# Applying the Peephole Metaphor in a Mixed-Reality Room



Andreas Butz  
University of Munich, Germany

Antonio Krüger  
University of Muenster, Germany

As introduced by Mark Weiser, ubiquitous computing describes a world in which mundane objects are equipped with processing power and networked with their environments to provide helpful services. Early ubiquitous computing research has concentrated on displays, with service provision dependent on both the user's<sup>1</sup> and the display's<sup>2</sup> location. While mobile computing devices have made interactions with single small displays commonplace, researchers continue to investigate interactions with large displays and multiple displays.

The introduction of IBM's Everywhere Displays projector<sup>3</sup> let researchers create displays on everyday surfaces and move them under the

environment's control. The original ED projector created stationary desktop interfaces in specific locations. In later applications, researchers used the technology more flexibly to label real-world objects and create visual elements in the physical world. It is possible to describe a steerable projector as a single, time-multiplexed room-sized display. All surfaces in a room can display information, though at a relatively low resolution and not necessarily at the same time. With-

in this display continuum, conventional displays can provide islands of higher resolution, better interactivity, and temporally unrestricted availability. This effectively adds a virtual layer on top of the physical environment, turning it into a mixed reality (MR) environment.

Given this, it might well be possible to one day create non-multiplexed room-sized displays using, for example, e-ink to create wallpapers and floor tiling. The environment could thereby simultaneously display visual elements on all of its surfaces. Thoughtlessly used, however, such an environment could easily become confusing and annoying by visually polluting otherwise calm surroundings. To prevent this visual pollution, we propose restricting visual display to small peepholes into the omnipresent and spatially continuous virtual layer. Here, we discuss this approach and how we implement-

ed the peephole metaphor using current steerable projector technology to create a room-sized MR environment. Within this environment, we developed several simple applications to demonstrate the peephole metaphor's viability.

## The peephole metaphor

The peephole metaphor's core idea is a virtual layer superimposed on a physical environment. While normally imperceptible, this layer can become visible, audible, or otherwise sensible when we open a peephole from our physical world into the virtual layer. Such a peephole might open as a display (visual peephole), a loudspeaker (acoustic peephole), or some other device. In a living room, for example, a display on a coffee table's surface might show users a collection of personal photographs.

The virtual layer has three basic properties: spatial continuity, temporal persistence, and consistency across peepholes. If we move two different portable display devices to the same position in our environment, they will display the same information (though not necessarily using the same graphical representation). We can implement a peephole through a tracked, head-mounted display (HMD), making the virtual layer a virtual world that is superimposed on the real world wherever we look, producing the general paradigm of spatial augmented reality (AR). Another option is to implement the peephole using a position-tracked handheld computer. This is similar to Ka-Ping Yee's peephole interfaces<sup>4</sup> and George Fitzmaurice's Chamaeleon,<sup>2</sup> both of which implement a spatially continuous virtual layer that is visible only through the peephole opened by the tracked display. Peepholes in our environment work much like magic lenses on a 2D screen. When we add interactivity, they behave like toolglasses.<sup>5</sup>

## Peephole types

There are two basic types of peepholes. User-initiated peepholes are actively created and manipulated by the user. System-initiated peepholes open through a display under the environment's control—such as a steerable projector. Such peepholes might direct users to certain content, provide alarms, or otherwise guide their attention to a particular position in space.

---

The peephole metaphor offers an interaction model for ubiquitous computing, organizing an environment's input and output facilities, such as sensors and displays.

We also distinguish between input and output peepholes. Output peepholes give users a view into the virtual layer, usually through situated (location-aware) displays. Input peepholes are similar to cameras, opening a view into the physical world from the virtual layer. An example here is a touch-sensitive surface. A location-aware touch-screen display therefore implements a two-way peephole that allows interaction (that is, input and output in the same place). However, input peepholes need not spatially coincide with output peepholes; obvious examples here are steerable cameras and steerable projectors mounted in (and pointing to) different locations.

### Technical aspects

Because they restrict simultaneous display, peepholes are ideally suited for time-multiplexed displays such as steerable projectors. To compute image rectification in a steerable projector, a *virtual camera* is moved synchronously with the physical projector.<sup>3</sup> The projector shows only what this camera records. Thus, when virtual layer items are visually represented in the 3D world surrounding the virtual camera, the physical projector creates a peephole into the virtual layer.

Tracked handheld displays, such as Chameleon<sup>2</sup> or the vampire mirror,<sup>6</sup> also directly create peepholes. They show a geometric 3D representation of the virtual layer, which in their case contains only 3D objects. This situation can be generalized for virtual layers containing arbitrary items, such as documents or media files. If we assign each item a certain spatial position, we can see (and manipulate) them on a tracked handheld display as soon as the display gets close enough to the item's position. The display's object representation doesn't need to be 3D—it can be a regular desktop icon, a full view of a text document or image, or any other visual representation suitable for the target display. If we're using a mobile phone as the display, it might display an icon or a text line with the object's name. If we use a PDA in the same location, it might display a preview version of the document, while a tablet PC might show us a full text or image representation that we can then edit or manipulate.

### Cognitive aspects

While traditional models of visual attention assume that everything we see is accumulated in a visual buffer that feeds all subsequent cognitive processing, newer models deny the existence of this continuously detailed buffer. Experiments have shown that visual disturbances in an image attract observers' visual attention and thus make them essentially blind to big image changes simultaneously taking place elsewhere. This effect, known as *change blindness*, provides strong arguments against a cognitive visual buffer that provides global, highly detailed information.

Rensink's model<sup>7</sup> explains how attention establishes visual coherence between objects. By guiding our atten-

tion to a certain object, we retrieve highly detailed visual information; we then lose this detail when we guide our attention elsewhere in the environment. Our memory retains a much coarser object representation, which allows us, for example, to remember the object's position in space.

This view of how visual processes are cognitively represented has striking similarities with the peephole metaphor. Rather than display all visual information continuously, we rely on guided attention to reestablish spatio-visual consistency between objects in the environment. In keeping with Rensink's model, we offer detailed representations of only those objects that users need for their immediate tasks. As a result, the instrumented environment—like the human brain—is able to save on resources when presenting and obtaining information to and from users. As Rensink's model suggests, users should be able to effectively interact with the virtual layer, even though only a small part of it is visible in detail.

Using peepholes, applications can maintain an explicit model of the user's attention, improving the match between the user's attention area and information location. Fur-

thermore, we can support simultaneous interaction of several users in the same instrumented space. We do this by using well-designed peepholes and by exploiting the cocktail-party effect—the human ability to concentrate on a specific area, despite disturbances caused by other individuals in the room.

### Interaction techniques

In a spatially continuous layer, moving objects between displays is logically reduced to moving the objects in space—if we assume a room-stabilized virtual layer and static displays. Spatial composition of displays thus comes for free. To take an object with us, we make it stick to a mobile display. If we release it somewhere else, we've transported it to that location. Objects attached to a display device change their spatial position synchronously with the device's position. Technically, with transportation, objects move from the global coordinate system into the target device's local coordinate system (its screen coordinates, for example).

By slightly extending the peephole metaphor, we can easily achieve scalability and file-access control. We do this using optical filters or curtains, which block or alter the appearance of whatever is behind them. If we think of situated displays as peepholes into a virtual layer, adding a filtering step is quite natural. Peepholes containing filters only show certain objects—such as objects based on some criteria or ones users are allowed to see. We can also combine filters and their effects. A logical **and** between filter effects places the filters sequentially; a logical **or** places them in parallel. Both placements are supported by corresponding visual hints. Conceptually, filtered peepholes can handle worlds with arbitrary

---

Peepholes in our environment work much like magic lenses on a 2D screen. When we add interactivity, they behave like toolglasses.

1 Sketch of our display continuum in the Saarland University Pervasive Instrumented Environment. Apart from the ceiling's steerable projector, it contains a projective desk, several displays of various sizes, and a set of speakers for spatial audio.



amounts of virtual layer data because they filter all but the relevant objects. The concept can also offer access control by providing, for example, read-only filters or existence filters (which show not what the object is, but only that it exists in a particular location).

### *Managing peepholes*

On the conceptual side, the peephole metaphor provides a terminology, a mental model, and an elegant way to describe different types of interactions in instrumented environments. On the technical side, a peephole manager can orchestrate interaction and ease application development. We've implemented a simple peephole manager in our instrumented environment.

As with a window manager, which takes care of window placement and input and output focus, the peephole manager organizes and administrates the environment's peepholes and their interactions. With peepholes and virtual objects, for example, the peephole manager can specify their positions and sizes in different frames of reference. At the room level, it places peepholes at 3D coordinates in the room's coordinate system. Because peepholes created by displays or surfaces are inherently 2D, it can also specify positions as 2D coordinates in the peepholes' local coordinate systems. The manager uses the same position specifications to specify content position in the environment's virtual information layer. While the virtual layer is superimposed and thus homomorphous to the physical 3D space, it can contain an arbitrary number of additional dimensions describing different information properties associated with a 3D room coordinate. One dimension, for example, might be ownership, while another is the data's temporal validity. We can then use filters to select the relevant subspaces, such as all files belonging to one owner, or all information less than a day old.

In addition to opening and closing static peepholes, the peephole manager can control gradual changes of

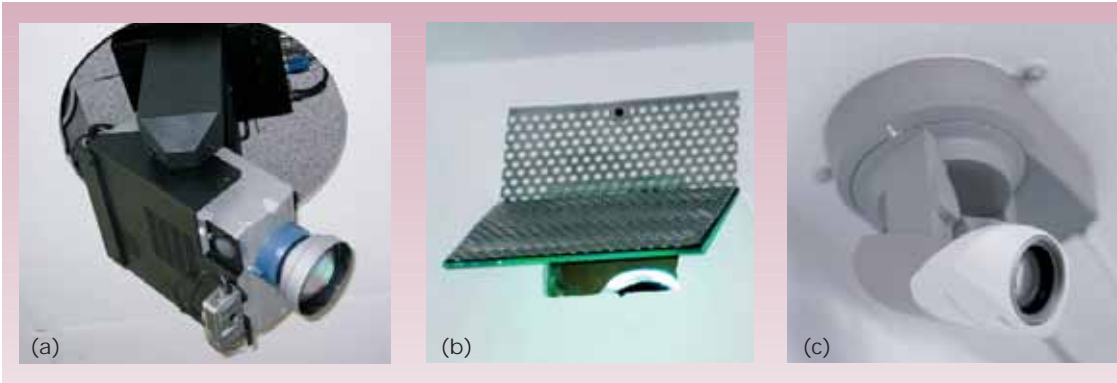
peephole properties over time (to move peepholes or recognize moving peepholes, for example). It does this through interpolators, rather than fixed coordinate values. If an object is positioned in a system-initiated peephole's local coordinate system and the peephole is moved across the room, the object will remain constantly visible as it travels from one place to another.

### A Display continuum and sample applications

To test and develop our ideas, we're using the Saarland University Pervasive Instrumented Environment (Supie), which provides a heterogeneous assembly of displays and sensors. Figure 1 shows a sketch of Supie and its available displays, which range from a fixed 50-inch plasma screen to a portable tablet PC and PDAs.

The room's central component is a ceiling-mounted steerable projector (see Figure 2a), which we can use to display information on every visible surface in Supie. The steerable unit is complemented by a fixed projector, which augments the office desk's surface. When designing Supie, we tried to blend all technology into the room's architecture so users could focus on interacting with objects rather than computing hardware. Although we could only partially integrate the steerable projector into the ceiling, we integrated the plasma screen into one wall and placed the fixed projector behind another, using a mirror to direct the projector's beam through a hole in the wall into the Supie (see Figure 2b). The overall installation creates a display continuum throughout the Supie, with varying physical properties (resolution and color depth) and temporal properties (the ability to display information at a specific time in a specific area of the room).

We've deployed several sensors to track portable screens and users in the environment. Four networked cameras (Figure 2c) track user positions, gestures, and optical markers. We use two corner-mounted cameras



**2** Display components. Among the main components are (a) a steerable projector; (b) a fixed projector, which aims its beam at the instrumented desk through a hole in the wall; and (c) steerable cameras.

that are steerable and have an optical zoom, which lets them focus on arbitrary locations. A third camera is located above the desktop surface and identifies objects and hand gestures. The fourth camera is on the steerable projector and facilitates user interaction with the information projected onto surfaces. We use active RFID tags to provide further information about users and tagged objects in the room. Although using RFID tag detection to track positions is rather coarse (1 to 3 meters), they help focus the cameras on room areas where objects or users are expected, thus supporting the optical tracking procedures.

In addition to the visual display continuum, Supie provides an audio continuum. The environment's eight-channel surround sound system lets us place sound sources at arbitrary room coordinates. This enables the environment, for example, to provide spatial audio cues that can guide users' attention to interesting areas within the visual display continuum. This is especially useful for user notifications when information appears outside the user's field of view.

Using these displays and sensors, we created several small applications in the SUPIE room, all of which use the peephole mental model.

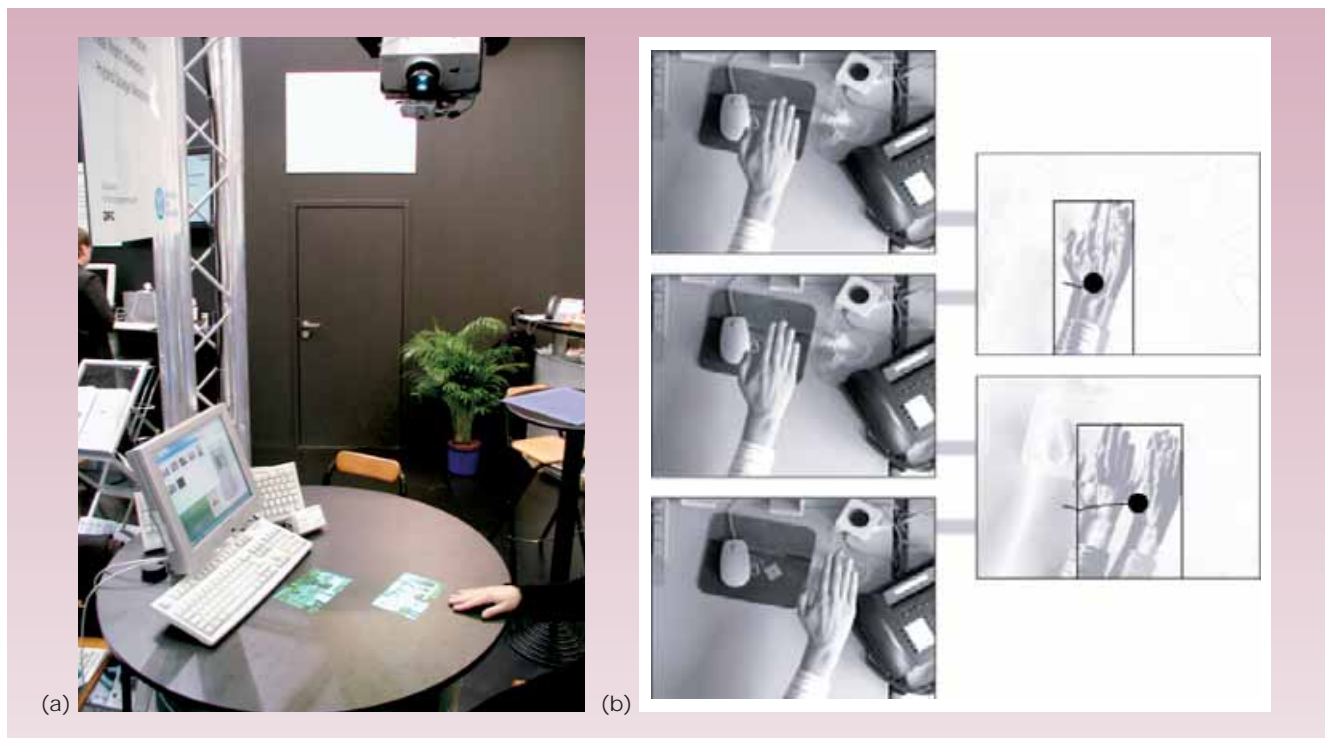
### *SearchLight*

SearchLight<sup>8</sup> was our first prototype to use system-initiated input and output peepholes. It implements input peeping to scan the environment for physical objects, and peephole output to guide user attention to specific objects.

In an initial scanning phase, the steerable projector's camera takes a series of overlapping pictures that cover the entire room. In these pictures, SearchLight uses the AR Toolkit library to find objects' optical markers, then stores their position in a database. This represents a system-initiated input peephole under the environment's control. Upon receiving a query, the environment actively highlights the target objects by opening a system-initiated output peephole in that location and projecting a bright spot around them (see Figure 3). Although we can implement this entire procedure without an explic-



**3** The SearchLight application. The system responds to query results by projecting a bright spot around the corresponding objects in the physical environment. (a) It finds one book on the shelf and (b) another lying on the windowsill.



4 The WipeIt application. (a) A steerable projector displays visual objects on a desk surface. (b) An attached camera uses difference images to determine whether a hand (or other object) moves across the frame in a consistent direction and recognizes this as a wiping gesture.

it 3D model of the environment,<sup>8</sup> we need a full spatial model to use other input technologies (such as RFID tags to track objects).

### *WipeIt*

Our wiping interaction technique aims to help users move objects to spatially distant displays. The steerable projector creates an output peephole to display objects on a desk surface (Figure 4a). The camera attached to the steerable projector opens the corresponding input peephole, observing object motions in its frame by analyzing difference images. If it recognizes an object (such as a hand) moving consistently at a certain speed and in a certain direction in the displayed object's location, it recognizes this as a wiping gesture (Figure 4b).

Users can wipe the objects from the desk surface in different directions. In our demo setup, the display closest to the wiping direction would catch the objects. They could thus be wiped either to the desktop monitor or to the white projection screen in the background (Figure 4a). The wiping gesture directly implements an ability to move objects to a different location in the virtual layer. Catching the object is important for visual feedback, and also makes the otherwise imprecise wiping direction and speed snap to a sensible value. Because objects are only wiped to other peepholes, they don't get lost in the virtual layer. WipeIt was our first prototype to explore interaction across peepholes.

### *Projecting object annotations*

Another example application we've built produces labels for physical objects. This function starts from the

3D model of a physical object and its subobjects, and then computes good and unambiguous locations for labels that explain object details.

The application places the labels—which can be text, graphics, or video objects—in the computed positions around the object (see Figure 5). The steerable projector then opens a peephole around the object and displays all annotation objects in the virtual layer. If the environment contains large amounts of virtual layer information, the peephole would naturally display any visible items near the annotated physical object. To prevent this, we used a filter with the annotation peephole so that it only displays annotation objects. The annotation application was our first prototype using filters in conjunction with peepholes.

### *Virtual room inhabitant*

One of ubiquitous computing's most pressing research problems is finding appropriate interface models for users to manage complex, behind-the-scenes machinery. In one of our prototypes, we investigated the use of a life-like anthropomorphic character that users can interact with.

Earlier studies with desktop systems have shown that interfaces incorporating life-like characters are subjectively rated better than standard interfaces.<sup>9</sup> We're convinced that life-like characters' advantages will multiply in instrumented environments with complex technical equipment. In such an environment, delegating tasks to a virtual character—also called a virtual room inhabitant (VRI)—seems like a natural way for users to achieve their goals (see Figure 6). The VRI can accompany the

user and provide personalized assistance by considering user preferences and available resources.

Conceptually, the VRI is a visual and audible manifestation of the environment's technology. From an implementation viewpoint, it's a special information item used to communicate planned or scripted content. The VRI makes itself visible by creating a system-initiated output peephole. As we describe elsewhere,<sup>10</sup> the VRI is projected through the Supie steerable projector.

The VRI can guide user attention to other peepholes for specific viewing situations. Because it can move freely through the environment, the VRI can, for example, signal users to follow her to a large, high-resolution display to view detailed content. This effectively represents a peephole switch: We now expect users to pay attention to the new peephole opened on the large screen. It's therefore safe to close the initial VRI peephole and use the technical resources (in this case, the steerable projector) for other purposes (such as new peepholes).

### **Spatial audio notification**

The Supie instrumented environment contains a speaker setup through which we can create spatialized audio in real time, using an extension of vector-based amplitude panning. In so doing, we can position sounds so that they apparently come from a particular direction or move along a certain path.

As we describe elsewhere,<sup>11</sup> we've developed a user-notification scheme that embeds optional parts into a background music composition playing in the environment. Users choose an instrument or rhythmic pattern to serve as their personal notifier. The background music's core part plays without this instrument or pattern until information for the user appears. The user's instrument or pattern is then added to the mix, seeming to arise from the direction of the new information. This is an acoustic form of system-initiated peeping. The environment opens an acoustic peephole in the same spatial position as the new information. Because the notification sound is simply in the same position as the information item, it directs the user's attention to that item.

### **Ongoing and future work**

The peepholes notion appeared relatively early in our work, and has evolved over time. While this article offers a snapshot of what peepholes can currently do, we believe the metaphor can be extended by even more mechanisms and analogies.

### **Peephole detection**

One important technical question is how to retrieve possible peephole locations. For building prototypes, it's possible to explicitly model locations and manually maintain a consistent environment model. While this is plausible for test purposes and for simple instrumented spaces, it doesn't scale to technically complex and dynamic instrumented environments. Hence, environments should automatically detect possible peephole locations. We're currently looking at possibilities for easily detecting display locations in complex spaces. To this end, we're investigating a two-step procedure involving only the environment's displays and cameras.



5 The object-annotation application. The application labels a complex object's subparts and projects labels where they will be easily readable.

In a first step, the displays start to blink in a characteristic color and sequence so that nearby cameras can easily detect them. Once one or more cameras registers the blinking, they will zoom in on the display to retrieve higher-resolution images. In the second step, the displays present geometrical patterns of a given size and shape, thereby allowing the cameras to determine their relative location and orientation.

This scheme nicely corresponds to a two-tier social protocol used in human communication: If we want to communicate to a person who is currently busy with something else, we first knock, wave, or utter an indistinct sound to get his or her attention, and only then begin communicating. In an instrumented environment containing multiple cameras and displays, displays can follow this scheme to inform the environment of their own position. Because the procedure matches the



6 The virtual room inhabitant. The steerable projector projects the VRI on the wall next to a large plasma screen

## Related work

Technically, recent work on steerable projector-camera units—such as Crowley’s portable display screen—is relevant to our research. The PDS provides a tracked cardboard surface onto which content is projected such that it appears stabilized on the cardboard. This involves real-time tracking and rectification of the projected image; it basically emulates a very lightweight and bright display, which can be freely moved.

Conceptually, our work draws on established ideas and interface concepts from spatially aware displays,<sup>1</sup> as well as on the notion of magic lenses and toolglasses.<sup>2</sup> The latter provide peepholes through specialized views into a virtual layer. These views let users navigate in this layer by physically moving the lens, glass, or display. More recently, Ka-Ping Yee<sup>3</sup> nicely defined the notion of peepholes into a virtual layer by describing interaction techniques for a PDA display in an imaginary information plane. Baudisch kept the imaginary virtual layer visible in his work on focus and context displays,<sup>4</sup> creating an island of high resolution in a larger, low-resolution display.

Finally, our peephole metaphor description implies several important properties recognized in other work. In

his roomware work, Streitz described a family of displays integrated into furniture. Within this framework, two such displays can be spatially composed to form a larger, spatially continuous display. The importance of spatial continuity is also emphasized in Rekimoto’s work on multiple display environments.

## References

1. G.W. Fitzmaurice, “Situated Information Spaces and Spatially Aware Palmtop Computers,” *Comm. ACM*, vol. 36, no. 7, 1993, pp. 38-49.
2. E.A. Bier et al., “Toolglass and Magic Lenses: The See-Through Interface,” *Computer Graphics*, vol. 73, no. 27, 1993, pp. 73-80.
3. K.P. Yee, “Peephole Displays: Pen Interaction on Spatially Aware Handheld Computers,” *Proc. ACM Conf. Human Factors in Computing*, ACM Press, 2003, pp. 1-8.
4. P. Baudisch and N. Good, “Focus Plus Context Screens: Visual Context and Immersion on the Desktop,” demo paper, emerging technologies, Siggraph 2002; [www.patrickbaudisch.com/publications/2002-Baudisch-Siggraph02-FocusPlusContextScreens.pdf](http://www.patrickbaudisch.com/publications/2002-Baudisch-Siggraph02-FocusPlusContextScreens.pdf).

human social protocol closely, the environment might also use it to attract a user’s attention.

## Wormholes

When looking for ways to implement the ubiquitous computing vision, we typically encounter large environments with some sort of virtual layer. Peepholes provide a convenient way to structure our interaction with virtual layer information in our visible environment (a single room, for example). In a world with many such rooms, we might want to use a similar metaphor to structure interaction among distant rooms and to create communication channels between them. Physics offers a convenient metaphor with Einstein-Rosen bridges, popularized in science fiction as wormholes. In physics, wormholes create a transportation channel between regions that are distant in space or time. In a metaphorical sense, wormholes create a channel from our local environment to a distant one.

In their simplest form, wormholes connect an input peephole on one side to an output peephole on the other side, and vice versa. Practically, these could be a camera and a display on both sides, which then provide a bridge for video conferencing. If we add an acoustic wormhole, users in the two distant environments can see and hear each other and thus have a conversation. We have thereby connected the two physical environments.

If we want to connect the two environments’ virtual layers, we can build virtual wormholes that basically reflect the virtual layer content in a specific area on one side to a specific area on the other. We can use these virtual wormholes to implement shared workspaces that let users on both sides open peepholes to manipulate the same objects. We can then apply filtering mechanisms, creating a metaphor for access restrictions and privacy.

We’re currently setting up two additional instrumented environments in Munich and Muenster. We’re planning to connect them to our instrumented environment in Saarbrücken through various wormholes, and hope to get additional insights from these practical trials.

## Conclusions

While our metaphor makes good use of today’s technology—such as steerable projectors—it will also be helpful for structuring future room-sized displays. Such displays might overcome today’s technical limitations, and, in principle, might thereby allow simultaneous, ubiquitous information display. One major challenge will be the deliberate creation, control, and manipulation of peepholes by the user. In our current applications, we mostly use peepholes controlled by the environment or regular portable displays. To fully use peepholes in our daily environments, however, we’ll need universal interaction techniques or devices for peephole control, such as hand or finger gestures or specialized input devices. Although we hope to find such tools or techniques in the areas of gesture-based interaction or tangible user interfaces, what they will actually look like is still largely unclear. ■

## Acknowledgments

The German Research Council funded our work under the Fluidum young investigator award and the special collaboration project SFB 378. We thank our anonymous reviewers for valuable suggestions, and all of our colleagues and students—including Christoph Endres, Ralf Jung, Michael Kruppa, Christian Schmitz, Michael Schmitz, Michael Schneider, and Mira Spassova—for helping to implement various aspects of this work.

---

## References

1. R. Want et al., *The Parctab Ubiquitous Computing Experiment*, tech. report CSL-95-1, Xerox Palo Alto Research Center, Mar. 1995.
2. G.W. Fitzmaurice, "Situated Information Spaces and Spatially Aware Palmtop Computers," *Comm. ACM*, vol. 36, no. 7, 1993, pp. 38-49.
3. C. Pinhanez, "Using a Steerable Projector and a Camera to Transform Surfaces into Interactive Displays," *Proc. ACM Conf. Human Factors in Computing*, ACM Press, 2001, pp. 369-370.
4. K.P. Yee, "Peephole Displays: Pen Interaction on Spatially Aware Handheld Computers," *Proc. ACM Conf. Human Factors in Computing*, ACM Press, 2003, pp. 1-8.
5. E.A. Bier et al., "Toolglass and Magic Lenses: The See-Through Interface," *Computer Graphics*, vol. 73, no. 27, 1993, pp. 73-80.
6. Andreas Butz et al., "Enveloping Users and Computers in a Collaborative 3D Augmented Reality," *Proc. Int'l Workshop Augmented Reality (IWAR)*, IEEE CS Press, 1999, pp. 35-44.
7. R. Rensink, "Internal vs. External Information in Visual Representation," *Proc. Smart Graphics*, ACM Press, 2002, pp. 63-70.
8. A. Butz, M. Schneider, and M. Spassova, "Searchlight—A Lightweight Search Function for Pervasive Environments," *Proc. Pervasive*, LNCS 3001, Springer Verlag, 2004, pp. 351-356.
9. S. van Mulken, E. Andre, and J. Mueller, "The Persona Effect: How Substantial Is It," *Proc. Human-Computer Interaction Conference*, Springer Verlag, 1998, pp. 53-58.
10. M. Kruppa, M. Spassova, and M. Schmitz, "The Virtual Room Inhabitant," *Proc. MU3I: Multi-User and Ubiquitous User Interfaces Workshop*, 2005; <http://www.butz.org/events/mu3i-05/submissions/p02-Schmitz.pdf>.
11. A. Butz and R. Jung, "Seamless User Notification in Ambient Soundscapes," *Proc. Int'l Conf. Intelligent User Interfaces (IUI)*, ACM Press, 2005, pp. 320-322.



**Andreas Butz** is a professor of computer science at the University of Munich, Germany. His research interests include user interfaces for ubiquitous computing. Butz has a PhD in computer science from Saarland University. Contact him at [butz@ifi.lmu.de](mailto:butz@ifi.lmu.de).



**Antonio Krüger** is a professor of geoinformatics and computer science at the University of Muenster, Germany. His research interests include intelligent user interfaces, and generation processes and human-computer interaction with limited technical and cognitive resources. Krüger received a PhD in computer science from Saarland University. Contact him at [antonio.krueger@uni-muenster.de](mailto:antonio.krueger@uni-muenster.de).