KUNGL
TEKNISKA
HÖGSKOLAN

CID-179 ● ISSN 1403-0721 ● Department of Numerical Analysis and Computer Science ● KTH

Tessellating Boundaries for Mixed Realities

eRENA ESPRIT Project 25379 Workpackage 6 Deliverable D6.4

John Bowers, Boriana Koleva, Holger Schnadelbach, Steve Benford, Chris Greenhalgh, Sten-Olof Hellström , Michael Hoch

CID, CENTRE FOR USER ORIENTED IT DESIGN

John Bowers, Boriana Koleva, Holger Schnadelbach, Steve Benford, Chris Greenhalgh, Sten-Olof Hellström , Michael Hoch

Tessellating Boundaries for Mixed Realities

Report number: CID-179, eRENA ESPRIT Project 25379 Workpackage 6 Deliverable D6.4
ISSN number: ISSN 1403 - 0721 (print) 1403 - 073 X (Web/PDF)
Publication date: Sept 2000
E-mail of author: yngve@nada.kth.se

Reports can be ordered from:

CID, Centre for User Oriented IT Design
NADA, Deptartment of Numerical Analysis and Computer Science
KTH (Royal Institute of Technology)
SE- 100 44 Stockhom, Sweden
Telephone: + 46 (0)8 790 91 00
Fax: + 46 (0)8 790 90 99
E-mail: cid@nada.kth.se
URL: http://cid.nada.kth.se

Deliverable D6.4
Tessellating Boundaries for Mixed Realities

ABSTRACT

This deliverable contains two major pieces of work specifically associated with Task 6.4 within Workpackage 6. (1) A concept of *traversable mixed reality boundary* is articulated and innovative designs for devices and displays are presented so as to support participants as they cross between 'realities'. (2) Techniques for gesturally engaging with *interaction surfaces* are described as being of core relevance to supporting participants in electronic arenas. Throughout we have been concerned to critically reflect and evaluate the interaction techniques and specific devices we have developed. In most cases this has involved their use in demonstrators which have gone beyond laboratory-based presentation.

Document ID	D6.4
Type	Report
Status	Final
Version	1.0
Date	Sept 2000
Author(s)	John Bowers (KTH), Boriana Koleva, Holger Schnadelbach, Steve Benford and Chris Greenhalgh (NOTT), Sten-Olof Hellström (KTH), Michael Hoch (ZKM)
Task	6.4

Deliverable D6.4
Tessellating Boundaries for Mixed Realities

Preface

John Bowers
Royal Institute of Technology (KTH), Stockholm, Sweden

0.1. Document Overview

This document is one of the final deliverables arising from Workpackage 6 of the eRENA project of the i3 schema of the ESPRIT-IV research action of the European Communities. eRENA is concerned with the development of electronic arenas for culture, art, performance and entertainment in which the general citizen of the European Community might actively participate supported by advanced information technology. Within this general context, Workpackage 6 has been concerned over the last two years of research with how interaction within such events might be supported. The concern in this workpackage is with questions of navigation, display and device design as well as uncovering general principles for interaction design which are appropriate to electronic arenas.

This deliverable contains two major pieces of work specifically associated with Task 6.4 within Workpackage 6. (1) A concept of *traversable mixed reality boundary* is articulated and innovative designs for devices and displays are presented so as to support participants as they cross between 'realities'. (2) Techniques for gesturally engaging with *interaction surfaces* are described as being of core relevance to supporting participants in electronic arenas. Throughout we have been concerned to critically reflect and evaluate the interaction techniques and specific devices we have developed. In most cases this has involved their use in demonstrators which have gone beyond laboratory-based presentation. For example, one technique for realising a mixed reality boundary has formed part of an artistic installation/performance that has been presented to the general public at a variety of European sites. Similarly, some of the techniques for interaction surfaces discussed in this deliverable have been used by musicians in genuine live performance settings again at a number of concerts in Europe. Other interaction technologies we have worked on have been carefully scrutinised by media professionals who are independent of the eRENA project. In addition, several of the tools developed in this workpackage have been fed back to Workpackage 7 and employed in public demonstrators there.

0.2. The Interaction Requirements of Electronic Arenas

Since Year 2 of the eRENA project we have been guided by an image of an electronic arena as *deploying mixed reality technologies to create environments for potentially large-scale real-time participation in media-rich cultural events*. This has set a number of specific challenges for the work in Workpackage 6.

First, interaction techniques must be appropriate for mixed reality settings – settings where physical and virtual objects and spaces can coexist. It is necessary, therefore, to go beyond refinements in, say, navigation techniques for VR so as to investigate how these might coexist with interaction in a physical environment. To gain some purchase on these matters, we have worked with a concept of 'mixed reality boundary' whereby physical and virtual spaces are conjoined. At such a boundary an impression is given to those on one side as to the activity in and nature of the space on the other side. A large-scale networked assembly of environments can then be conceived of as a 'tessellation' of spaces of different kinds, some virtual, some physical.

Second, the workpackage has been concerned with the support of large-scale, real-time interaction. This means we have had to go beyond the traditional perspectives of HCI (Human Computer Interaction) research which tend to focus primarily on the interaction experience of a single user working with a single application on an isolated workstation. All of the interaction techniques developed in this workpackage have been concerned to support the activities of users in a multi-participant social environment. This has required us, from time to time, to ensure that our solutions 'scale-up' as well as investigate techniques specifically for supporting the mass action of a collectivity. We have also been cognisant of the fact that participation in an event in an electronic arena can take many forms. There can be actors, observers, production support personnel, and so forth. Commonly, different kinds of participant need some awareness of what others are doing. This means that in a participatory environment, one often has to design interaction techniques with *'third parties'* in mind. That is, the engagement of one user with their technology has to be designed in such a way that others can pick up on what they are doing. This is particularly clear for technologies designed for performers but has turned into being a general theme of much of what we report here.

Third, electronic arenas are conceived of as settings that are media-rich. That is, an electronic arena should constitute a rich and lively 'sensorium' for its inhabitants. This can introduce tensions, however, as we wish to enable such environments to be as participatory as possible. How can participants, typically drawn from the general citizenship, engage effectively with complex media-rich dynamic mixed realities? Researchers in this workpackage have been concerned throughout the lifetime of eRENA to investigate strategies for, as-it-were, 'algorithmically amplifying' everyday non-virtuosic human gestures so that ordinary folk can influence or create complex media presentations. On this theme, this workpackage presents some examples of algorithmic enhancement of simple gesture in the context of interactive electronic music performance.

In all these respects, research in this workpackage has had to address quite specific requirements related to our concerns for electronic arenas. However, it has done this in such a way as to produce work of relevance in the general context of HCI research. Indeed, material from both the major contributions to this deliverable have already been published and demonstrated at the CHI2000 conference of the Association of Computing Machinery (ACM) held at The Hague, Netherlands this year.

0.3. Structure of this Document

This document is structured as follows. After this Preface, there follow two chapters.

Chapter 1 describes the work conducted at Nottingham on traversable interfaces. Traversable interfaces are intended to establish the illusion that virtual and physical worlds are joined together and that users can physically cross from one to the other. The design for a traversable interface presented in this chapter combines work on tele-embodiment, mixed reality boundaries and virtual environments. It also exploits non-solid projection surfaces, of which four examples are noted. The design accommodates the perspectives of users who traverse the interface and also observers who are present in the connected physical and virtual worlds, an important consideration for the performance and entertainment applications of core interest in eRENA. A demonstrator is described in this chapter which supports encounters between members of a laboratory and remote visitors. This adds to the practical instantiation of these concepts to be found in the work on mixed reality performance in Workpackage 7 of the project.

Chapter 2 presents work by KTH in collaboration with the ZKM. The chapter falls into two halves united in their concern to work out principles for the design and use of *interaction surfaces*. It is argued that supporting human gestural interaction in relation to a surface is consistent with Chapter 1's concern for boundaries. Although a surface in a 3D environment is typically a flat 2D affair, this does not mean that gestural interaction in relation to it need be impoverished or lacking in expressivity. A number of interaction techniques, motivated by design principles derived from a consideration of the nature of electronic arenas, are proposed for enabling activity at surfaces to control complex interactive media. In the first half of the chapter, examples are given to support simple gestural engagement with synthesised sound. The specific technologies described in this chapter for sound control have been successfully used by performers of improvised electroacoustic music at a number of European performances. Some avenues for their future enhancement are also indicated. The second half of the chapter describes the general motivation behind the construction of the RoundTable tangible interface which figured so prominently in Deliverable D4.5, This again supports gestural interaction in relation to a surface – this time mediated by the manipulation of blocks placed on top of a computer graphical projection. The hardware and software construction of the RoundTable is described, together with an overview of experience using it (with more specific accounts being available in Deliverable D4.5).

Taken together the two chapters give a strong image of how interaction can be supported in a mixed reality arena composed of a tessellation of physical and virtual locales linked by traversable boundaries which embody innovative display surfaces and interaction techniques.

0.4. Relationship of this Document to the eRENA Workplan

This document is one of the two final deliverables from Workpackage 6, the other (D6.5) gathers together the work of the GMD in a single report.

In the eRENA workplan, Workpackage 6 has been composed of five work tasks. This is the deliverable associated with Task 6.4. In the Year 3 amendment to the eRENA project programme document, Task 6.4 was redefined to commit its partners to accomplishing the main outcomes described in the following (text slightly reordered for clarity):

1. “This task will consider the different kinds of boundaries that may be established between the real and the virtual and will also consider how multiple boundaries can be used to join together (‘tessellate’) many different physical and virtual spaces into an integrated environment. Nottingham will further elaborate its notion of 'mixed reality boundaries' explored in Workpackage 7b to give coherence to this task. We will explore how the general concept of boundaries between the physical and the virtual (which may have a variety of properties) can provide a unifying conceptual framework for all eRENA's work on mixed reality. A number of interfaces will be constructed to instantiate our concepts. Nottingham will build upon the use of the ‘rain curtain’ technology from Workpackage 7b in Year 2, generalising it to create a variety of designs for traversable interfaces between the physical and the virtual, appropriate to different situations. Nottingham will also consider how multiple boundaries might be used to join many physical and virtual spaces together into a unique mixed reality structure.”
2. “Another aspect of this Task's research will be understanding how to map high dimensional data (required, for example, to specify computer graphical animation or synthesised sound) to lower degree of freedom devices which are readily usable without requiring a high degree of expertise or excessive dexterity or training. KTH will elaborate the concept of 'algorithmically mediated interaction', developed in Years 1 and 2 of the project, to further guide eRENA's treatment of these topics. KTH will concern itself with the design of interactive surfaces and regions that can be embedded within a mixed reality environment. This will include refining principles for interaction design with 2D contact surfaces (e.g. touch pads) as well as non-contact, free-gestural methods (e.g. video analysis). In collaboration with KTH, ZKM will develop a room-sized mixed reality environment containing a variety of tangible interfaces. Here, video-tracking techniques will be investigated to enable the manipulation of physical objects to control views on virtual environments and mix sound sources. The target application of production for inhabited television will give a specific focus to our general exploration of tangible interfaces and provide points of integration with Workpackages 4 and 7a.”
3. “Drawing on work in collaboration with BT in Task 6.3 in Year 2, KTH will further develop and evaluate video-based interaction techniques to support unencumbered participation in mixed realties. This will complement the work at ZKM by examining video analysis techniques which do not involve tracking objects in the display.”

Our work in eRENA has addressed all of these and only a few small words of clarification are needed.

Commitment 1, concerning traversable mixed reality boundaries and their tessellation, has been met precisely as specified in the project programme. The outcomes of this work form Chapter 1.

Commitment 2 exactly refers to the work done at KTH on touchable interaction surfaces and reported in Chapter 2. It also picks out the work KTH and ZKM have jointly achieved with the RoundTable and how it can be embedded within a room-sized mixed reality environment (see also Deliverable D4.5). This is also reported in Chapter 2 here.

Commitment 3 has also been attended to. BT and KTH have indeed collaborated on video-based interaction techniques as described. These support non-contact, free-gestural interaction and the capturing of movement data from potentially a large number of simultaneous participants in an electronic arena. The technology implemented (bFinder) had a key role in one of the demonstrators in Workpackage 7. For completeness of presentation, it is described there (in Deliverable D7b.4) rather than in the current deliverable.

Accordingly, we hold that partners collaborating on Task 6.4 have completed its associated work and deliverable with little relevant deviation.

Chapter One
Traversable Interfaces Between Real and Virtual Worlds

Boriana Koleva, Holger Schnadelbach, Steve Benford and Chris Greenhalgh
University of Nottingham, Nottingham, UK

1.1. Introduction

Various technologies have been developed to allow people to experience remote environments. These might be virtual environments that are experienced through virtual reality technologies or physical environments that are experienced through tele-embodiment and tele-presence technologies. A thread running through this research is the idea of using immersive technologies to establish the illusion of entering the remote environment, resulting in a sense of presence.

A major weakness in this illusion is that users clearly do not leave their physical environment behind them when they enter a remote environment. They remain firmly and visibly present within their local physical space. This is a problem for two reasons. First, their own illusion of remote presence may be destroyed by distractions from the local physical space. Examples can be found in previous experiments with virtual reality. In studies of presence in single user virtual environments, users reported that 'breaks in presence' were caused by background noise and interference from hardware such as cables (usoh et al., 1999. Bowers *et al.* (1996) note how conduct in a collaborative virtual environment was disrupted by events in the physical environments of the participants. Second, observers of the interaction can clearly see that participating users have not gone anywhere. This is a particular problem if the interaction is being staged at least in part for the benefit of these observers, for example as part of an entertainment or performance application of the sorts of core interest in eRENA. It might also be a problem if these observers may themselves become participants at a later date. For example, if they are waiting their turn in an entertainment application or in a shared working environment due to the limited availability of equipment.

Our response to these problems is the concept of traversable interfaces. These enhance the illusion of immersion by making it appear that participants leave their local physical environment in order to enter into a new remote environment. They aim to do this in a way that makes sense to the participants who are entering the remote environment, to observers who are already in the remote environment, and to observers who remain behind in the local physical environment. Our discussion will focus on traversal between physical and virtual environments. However, a traversable interface could also be used in a tele-presence application to link a local physical environment to a remote physical environment.

Further motivation for traversable interfaces is provided by recent work on mixed reality. Paul Milgram has classified mixed reality technologies according to a 'virtuality continuum'(Milgram and Kishino, 1994). At one extreme of this continuum we find purely physical environments and at the other purely virtual environments. In between, we find augmented reality where physical environments are enhanced with digital information, and augmented virtuality, where virtual environments are enhanced with physical information.

Traversable interfaces provide a mechanism for people to dynamically relocate themselves along this continuum. At one moment they may be primarily located in augmented reality, with a view into an adjoining virtual environment. They may then traverse the interface and find themselves primarily located within an augmented virtuality, with a view back into a physical environment. Traversal allows people to move back and forward between primarily real and primarily virtual environments, repositioning themselves along the virtuality continuum, according to their interest and whether they want the physical or virtual to be their primary focus.

The rest of this chapter describes our general notion of traversable interfaces and the different manifestations we have experimented with. This chapter, then, generalises the approach to mixed reality boundaries first reported in Deliverable D7b.1 and also instantiated in D7b.3, while relating it to general human-computer interaction issues.

1.2. Traversable Interfaces

We begin with a general design for a traversable interface. Figure 1.1 summarises the illusion that we wish to create. On the left we see a physical environment that is connected to the virtual environment on the right. Our design needs to consider the perspectives of the four classes of participant, A, B, C and D. A is an observer in the physical environment. B is an observer in the virtual environment. C is crossing from physical to virtual, and D is crossing from virtual to physical.

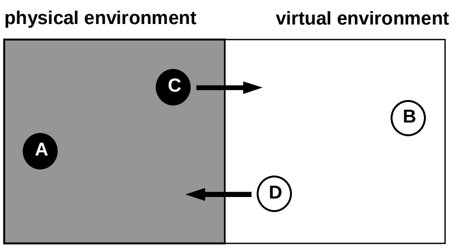


Figure 1.1: the illusion that we wish to create

An important point is that the illusion should potentially work for all of these classes of participants, although some applications may give priority to one class over another. For example, a performance might require that the audience believe the illusion, while the performers could be aware of the mechanisms involved. This observation challenges traditional approaches to interface design that have focussed on the experience of the direct participant, but have tended to neglect the experience of observers. We suggest that this is an important consideration for any application where an interface is deployed in a shared or “public” environment, including office environments as well as performance and entertainment applications.

Two other general points should be noted. First, objects as well as participants might traverse the interface. Second, partial traversal might be possible, for example pushing a limb through the interface. However, in this chapter we restrict our consideration to complete traversal by humans.

Our general design for a traversable interface integrates a number of techniques:

- mixed reality boundaries (Benford et al., 2000) for creating windows between physical and virtual environments.
- tele-embodiment for allowing remote virtual participants to enter a physical environment (Kuzuoka et al., 1995; Paulos and Canny, 1998).
- immersive interfaces for accessing virtual environments, including head-mounted displays (HMD) and projected displays ranging from single screens up to multi-surface CAVEs (Cruz-Neira et al., 1992).
- non-solid projection surfaces to allow participants to seemingly pass through a projected image, moving from a public to a more private physical space.

The following sections describe how these are integrated into an overall design, beginning with the idea of mixed reality boundaries.

1.2.1. Mixed reality boundaries

Mixed reality boundaries represent a specific approach to mixed reality that involves creating transparent windows between physical and virtual environments so that occupants of each can communicate with the other (Benford et al., 2000). In contrast to other approaches that focus on superimposing the two environments on top of one another (e.g., augmented reality typically overlays a virtual environment on top of a physical environment), the spaces on either side of the boundary are adjacent, but remain distinct. A feature of this approach is that multiple boundaries might be used to join together many different physical and virtual environments into a larger mixed reality structure.

Figure 1.2 shows how a simple mixed reality boundary can be created. On the left is a physical environment and on the right a virtual environment. An image of the virtual environment is projected into the physical environment and an image of the physical environment captured from a video camera is displayed as a live video texture within the virtual environment. The physical and virtual cameras and projections are aligned so that the images appear to be the reverse sides of a common boundary.

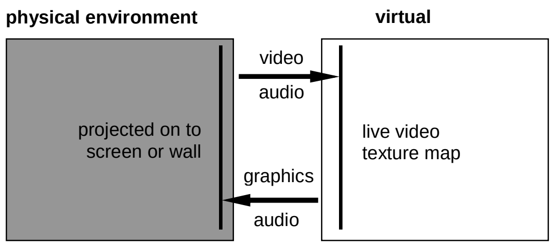


Figure 1.2: a simple mixed reality boundary (from Benford et al., 2000)

A variety of mixed reality boundaries might be created with different properties in terms of their ‘permeability’, the extent to which they allow information and objects to pass across them; ‘situation’, their spatial relationship to the connected spaces; ‘dynamics’, their temporal properties; and ‘symmetry’ (Koleva et al., 1999). Permeability properties are particularly interesting here because they include the sub-property of ‘solidity’, the extent to which a boundary allows objects and participants to pass through it. This can be broken down into two issues, how to allow participants and objects to enter the remote environment and how to create the illusion that they have left their current environment when doing so.

1.2.2. Entering the remote environment

Entering a remote physical environment can be achieved by taking control of a remote physical proxy such as a robot. The field of tele-robotics is well established, particularly in areas such as working in hazardous environments such as outer space and the deep ocean. Of more direct relevance here is recent work on tele-embodiment in collaborative settings, where participants take control of a physical proxy or surrogate (Kuzuoka et al., 1995). In one recent example, participants control a tele-embodiment called a Personal Roving Presence (PRoP) that is armed with a video camera, microphones and speakers, and steer it round a remote environment in order to meet and converse with others (Paulos and Canny, 1998). Designs for early PRoPs include ‘space browsers’, helium filled blimps that act as airborne tele-robots and ground based platforms called ‘surface cruisers’. By placing a PRoP on the physical side of a mixed reality boundary and integrating the controls for this PRoP and the video and audio from it within the virtual environment, participants on the virtual side could enter the physical.

An alternative approach towards introducing remote virtual participants into a physical environment would be to use shared augmented reality technology such as (Billinghurst et al., 1996). See-through HMDs could display avatars superimposed onto the physical scene. In fact, this could be combined with the use of PRoPs. The position of the PRoP could be tracked and the image of the avatar superimposed upon it.

Techniques that allow a user in a physical environment to enter a remote virtual environment are well known and include a range of immersive displays including HMDs and different tracking and interaction mechanisms for interacting with a projected image of a virtual environment.

1.2.3. Leaving the current environment

The illusion of traversal requires that a user is seen to leave their current local environment when they enter the remote one. We propose that this may be achieved by using non-solid projection surfaces so that the user can appear to directly step into and through the image of the remote environment.

This is straightforward in the virtual environment. The image of the remote physical environment is displayed as a video texture attached to a graphical object. This can be non-solid, enabling avatars to pass through it.

It is more difficult in the physical environment. Later on, we shall describe four different approaches that we have implemented involving projection onto non-solid materials such as water, the use of fabric curtains as well as mechanical devices such as sliding doors and movable screens. For the remainder of this section we shall assume the existence of such technologies.

It should be noted that in all cases, what actually happens is that the user passes from a public space through the image, into a more private space beyond. From the physical environment they move to a physical antechamber beyond the screen where they find the immersive technology required to enter the virtual environment. From the virtual environment, their avatar moves to a virtual antechamber beyond the screen where they may find the controls to access a PRoP. The physical antechamber may take on a variety of forms. In a performance, the public space will be the key focus of activity, with the antechamber being ‘the wings’ or behind the scenes. Conversely, the antechamber might be the main focus of the activity, for example it might be a CAVE installation (Cruz-Neira et al., 1992), with the traversable interface providing an entry point to and from the outside world.

1.2.4. An integrated design

Figure 1.3 shows how the above techniques for entering and leaving physical and virtual environments can be integrated into a general traversable mixed reality boundary.

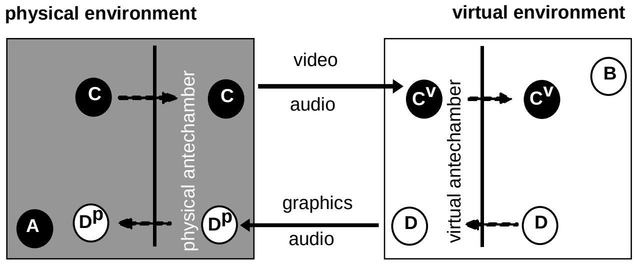


Figure 1.3: creating a traversable boundary

On the left is a physical environment containing a non-solid projection surface onto which is projected a view of the remote virtual environment. Behind this is an antechamber containing the immersive technology required to become embodied within the virtual environment. On the right is a virtual environment containing the video view into the physical environment. Behind this is a virtual antechamber that contains controls for a remote PRoP and that also contains a second video texture showing the view from this PRoP as it moves around the physical environment.

We can now consider how the four participants A, B, C and D from Figure 1.1 will experience this design. Participant A is the observer in the physical environment. They will see participant B through the mixed reality boundary. They will see participant C step through the physical projection screen, apparently into the virtual world. At the same time, they will see C's virtual avatar, C^V emerge into the virtual world. They will see participant D's avatar approach the projection screen and then disappear from view. D's ProP, D^P, will then emerge through the physical screen.

Participant B is the observer in the virtual environment. They will see A through the boundary and will see C approach them in the video view, disappear and then replaced by C's avatar, C^V, appearing through the video texture. They will see D's avatar approach the video texture, pass into it and then see D's physical proxy, D^P, appear in the video image.

Participant C traverses from the physical to the virtual. They will step through the physical projection screen, entering the physical antechamber. There they will find the technology required to independently access the virtual environment. This might be a headmounted display, desktop computer, CAVE, specialised vehicle (for example, a 'pod' in a simulation ride) or further projected display. Their avatar will initially appear in the virtual antechamber and they will then steer it through the video texture into the public virtual environment.

Participant D traverses from the virtual to the physical. They will steer their avatar through the video image of the remote physical environment, entering the virtual antechamber. Here they will find the virtual controls for the remote PRoP, D^P, as well as a further video texture showing the view from its onboard camera. They will then be able to steer the PRoP from the physical antechamber, through the physical projection surface into the public physical environment.

1.2.5. Design considerations

This design for a traversable interface is a general one. A particular realisation will have to make a number of specific design choices in order to meet the two goals of traversable interfaces as outlined in the introduction.

The first goal was to minimize distractions for participants who wish to become present in a remote environment. This is achieved by locating the VR equipment required to access this environment in a private antechamber. This can be designed to provide an optimal operating environment for this equipment, for example, being painted and lit to support video tracking, being free of other equipment that might interfere with electromagnetic tracking, and generally being free of clutter on which the user might snag themselves.

The second goal was to create the illusion of physically leaving the current environment in order to enter a new remote environment. Successfully meeting this goal will require considering the following design issues.

The physical and virtual antechambers can be decorated to support the transition to the new environment. For example, in a theme park ride, the physical antechamber might be modeled to match the virtual world. If the user thinks that they were going to pass into a virtual cave, then this antechamber should look like that cave. The physical and graphical design of PRoPs and avatars can also support the illusion of traversal. In a theme-park ride, the PRoP might be a sophisticated animatronic figure (such figures are already used in theme-parks). Likewise, the positions of physical bodies, PRoPs and avatars at the key transition points will be important. With careful design, it may be possible to make them appear to directly replace one another, to be overlaid on one another, or to time the sequence of appearances and disappearances to reinforce the illusion of traversal.

Traditional theatrical techniques may be used to enhance or alter the illusion of traversal, including changes in lighting, the use of smoke and sound effects. Another key effect is the use of shadows. Several of the non-solid projection surfaces that we introduce below can be configured to show the physical user beyond the screen as shadow. In some cases it will be important to avoid shadows so as to maximise the illusion of traversal. In others, the silhouette of a participant’s body seen against the image of the virtual environment may be used for its artistic effect (see Figure 1.8) or as one way of overlaying participants’ physical and virtual bodies as noted above.

1.3. Designs for Non-Solid Projection Surfaces

The use of non-solid projection surfaces is an essential part of our design. It has also been the most challenging part to realise. This section describes four attempts to construct such surfaces: fabric curtains, water curtains, a sliding door, and a flip-up screen. Figure 1.4 summarises the four designs and shows examples of each.

1.3.1. Fabric curtain

Curtains are familiar devices for partitioning physical space. Curtains can provide privacy and can be readily traversed, introduced and removed. They have been extensively used in theatre to hide and reveal actors and objects and to give the illusion of transitions between scenes. There are a wide variety of familiar designs of curtains; they can be pulled back, raised, vertically slit and be formed into blinds.

Curtains can be made of materials that can hold a projected image and so represent a natural choice for creating non-solid projection surfaces. Our initial design as shown in Figure 1.4 (a) is based around a number of vertical segments of projection screen fabric, weighted at the bottom to hold their shape. A user can easily push through these and the curtain settles down to its regular shape within a few seconds. We back project the image onto the curtain by bouncing it off of a mirror on the ceiling. This creates an area in the antechamber where a participant may stand or sit without casting a shadow onto the screen. Conversely, they may be deliberately positioned so as to create a shadow for artistic effect as noted above. Figure 1.4 (b) shows a participant emerging through the curtain.

1.3.2. Water curtain

Over the last two years in eRENA, we have also extensively experimented with a second curtain – a curtain of water. In 1998 we began collaborating with the performing arts company Blast Theory who were already experimenting with projecting images and video into a vertical curtain of water. Projection into water has also been explored in other contexts. For example, Disney-MGM studios projected film clips into fountains and a water screen as part of a dream sequence in their “Fantasmic” show in their October 1998 program.

The overall design of the water curtain is shown in Figure 1.4 (c). The curtain is produced by several fine spray nozzles (originally designed for spraying pesticide) attached to a metal pipe that is suspended roughly two meters above a trough on the ground. Water is pumped through the pipe, descends as a fine spray about half a meter thick and is collected from the trough and recycled. Figure 1.4 (d) shows this physical infrastructure. The water curtain holds a back-projected image surprisingly well, although early experimentation showed that the projector needs to point straight at the curtain, making shadows unavoidable as participants pass through it.

Being completely fluid, a person or object can pass through the water curtain much more seamlessly than they can with a fabric curtain (so long as they are prepared to get wet!). It is also transparent when viewed from behind, allowing for easy observation of its users (e.g., by performers who can then time their emergence through the curtain to match the user’s actions). Like a fabric curtain, the water curtain can be readily introduced and removed by switching the pump on and off. Holes can be dynamically punched through it by using solid objects to interrupt the flow of the water. Finally, it has a powerful aesthetic, in terms of the continually shifting quality of the visual image, the sound of the water and its physical feel.

In January 1999 we staged a public demonstration of using a water curtain as an interface to a virtual environment (see Deliverable 7b.1). Participants undertook a journey through a virtual world, during which they were interrupted by a performer emerging through the curtain – an event that had a significant theatrical impact. Figure 1.4 (e) shows the performer emerging through the water curtain. More recently, we developed the full-scale public performance environment that involved the use of six rain curtains to allow an audience to experience a shared virtual world (see Deliverable D7b.3).

1.3.3. Sliding door

Unlike a curtain, a door is a solid projection surface that is traversed by physically moving a large section of it. As with conventional doors, there are many potential designs including hinged, sliding and rotating. Our first design has been a sliding door made from perspex as shown in Figure 1.4 (f). Figure 1.4 (g) shows a participant opening the door in order to step through it.

The sliding door has several interesting properties. Being solid, it can more easily be locked than a curtain, allowing participants to minimise possible interruptions. Its solidity also favours applications where it is part of a more permanent architectural framework. Early tests suggest that our sliding door can simultaneously hold two different images, one on each side, provided that the images that have similar contrasts (otherwise there is too much visible interference between the two). This potentially saves space, as it only requires one projection surface to display both public observers' and immersed participants' views of the virtual environment. The properties of solidity and holding multiple images could usefully be combined in using a sliding door as the entrance to a CAVE. One surface of the CAVE could be slid open to allow participants to enter. Visitors remaining on the outside could see a specially tailored (e.g., without head-tracking) public view of the activity in the CAVE on the outside of the door.

1.3.4. Flip-up screen

Our final example is a flip-up screen as shown in Figure 1.4 (h). This is a screen that can be moved from a vertical to a horizontal position at ceiling height, allowing people to pass underneath it. Figure 1.4 (i) shows a participant raising the screen. The flip-up screen is essentially a specialised form of door. However, it has the additional property of being able to act as an ambient display surface when in the raised position, reflecting the idea of ambient display media proposed in (Ishii and Ulmer, 1997). This is possible because the projected image is bounced off of the mirror on the ceiling and hits the screen when it is in both its vertical and horizontal positions.

This property suggests an alternative mode of use to the previous examples. Instead of stepping through the projected image, the user may remain in one physical location, but choose to lower or raise the flip-up screen according to whether their interaction is primarily focussed in the physical or the virtual environment. To focus on the physical environment, the user raises the screen, opening up their physical space to the public space beyond and displaying a peripheral image of the virtual environment on the ceiling. Figure 1.4 (j) shows a participant who is focussed on a task in the physical world and so has set the flip-up screen to its ambient position. To focus on the virtual environment, they place the screen in its vertical position, shielding their local physical environment from the public space beyond, and providing users in this public space with an image of their avatar in the virtual environment instead of their physical self using the immersive technology. In this way participants can reposition themselves along Milgram's virtuality continuum as noted in the introduction.

An extension to this approach would be to use the physical raising and lowering of the screen to drive a switch to automatically configure a user's local environment according to whether they were currently in the physical or virtual environment. The switch might configure lighting and tracking technologies and might minimise distractions, for example by routing the user's phone to their voice mailbox when they were immersed in the virtual environment. This reflects previous work on using physical doors to manage electronic privacy in an office environment, using a so-called "doormouse" (Buxton, 1995).

In summary, we have realised four different kinds of non-solid projection surface that might be used in traversable interfaces. These can be broadly grouped into the two categories of curtains (fabric and water) and doors (sliding and flip-up). The curtains potentially offer the most seamless illusion of traversal and could be especially suited to performance, art and entertainment. The doors provide a less fluid illusion of traversal, but may offer some practical advantages for use in everyday environments such as offices and the home. Of course, there are many other possibilities. Perhaps we can use other materials such as smoke to create highly fluid projection surfaces, and no doubt there are other possible mechanical designs based on doors and curtains.

1.4. Demonstrations

We have developed a demonstration of a traversable interface in order to show that our design is technically implementable. It should be noted that we do not claim that this is yet a real or effective application, although our future plans involve developing and evaluating such an application.

Our demonstration has been constructed in our laboratory. Its aim is to provide a social space where lab members can meet with visitors who "drop in" over the Internet. A mixed reality boundary allows lab members and visitors to see and talk to one another. Both can also traverse this boundary. A single visitor at a time can take charge of a simple PRoP and use it to explore an area of the laboratory. A single lab member at a time can step into the virtual world to become part of a virtual meeting. Figure 1.5 shows the collaborative virtual environment that we are using in our demonstrator. This has been realised using the MASSIVE-2 system (Benford et al., 1997). The image shows the video texture that forms half of the mixed reality boundary with the physical environment.

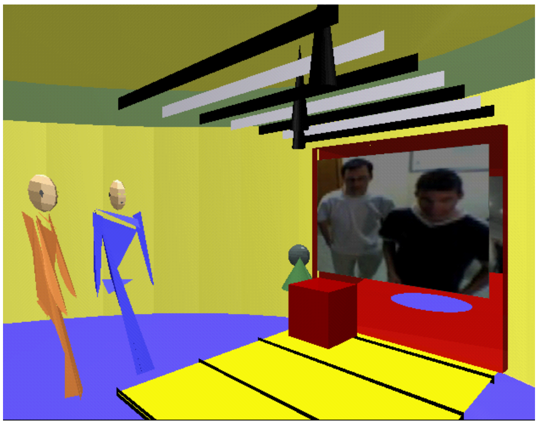


Figure 1.5: the virtual environment with video texture

Figure 1.6: controlling the PRoP from MASSIVE-2.

Avatars can step through this boundary to enter a small virtual antechamber where they find the interface to control our remote PRoP. This consists of a second video texture that shows the view from the PRoP's on board camera as well as six buttons, four to move the PRoP forwards and backwards and to rotate it left and right, and two to tilt the camera up and down vertically. Figure 1.6 shows the view over a remote user's shoulder when they have just entered this antechamber. Inset is a close up of the virtual controls for the PRoP.

The PRoP itself is a small wireless robot that has been constructed using a LEGO Mindstorm kit (see Figure 1.7). This platform can be moved around the floor, includes a raisable arm for the camera and can be controlled over an infrared link. A small wireless video camera and microphone have been mounted on the platform along with a pen-torch to illuminate nearby objects. The wireless connections currently have a limited range and there are as yet no on-board speakers (so the PRoP can see and hear, but not talk). The PRoP is also rather small, standing at approximately one foot tall. However, it does provide an inexpensive workable solution for initial demonstrations and application development.

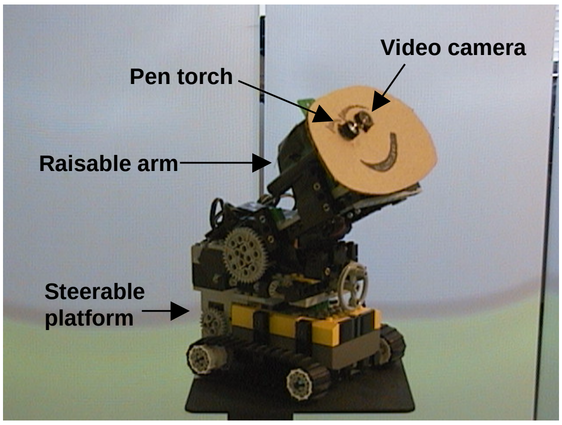


Figure 1.7: the PRoP

Figure 1.8: immersed in the virtual environment

The physical side of the boundary can utilise the fabric curtain, sliding door or flip-up screen designs. The images in Figures 1.4 (b), 1.4 (g) and 1.4 (j) all show examples of the view looking into our virtual environment, as if from out of the video texture. In each case a video camera is mounted on the top of the frame of the boundary to provide the video view shown in Figure 1.5.

This positioning is less than ideal as the two sides of the boundary are not strictly spatially aligned and a solution that allows a small camera to be located in the centre of the screen is ideal. Mounting the camera in the centre of the flip-up screen would also allow it to provide a peripheral view from above the user's workspace when in the raised (ambient) position. Having traversed the physical screen, the user enters the physical antechamber and finds equipment to access the virtual world. Figure 1.8 shows an example where the user has stepped through the screen and has donned a head-mounted display. In this case, they have been deliberately positioned so that we see their shadow.

1.5. Summary

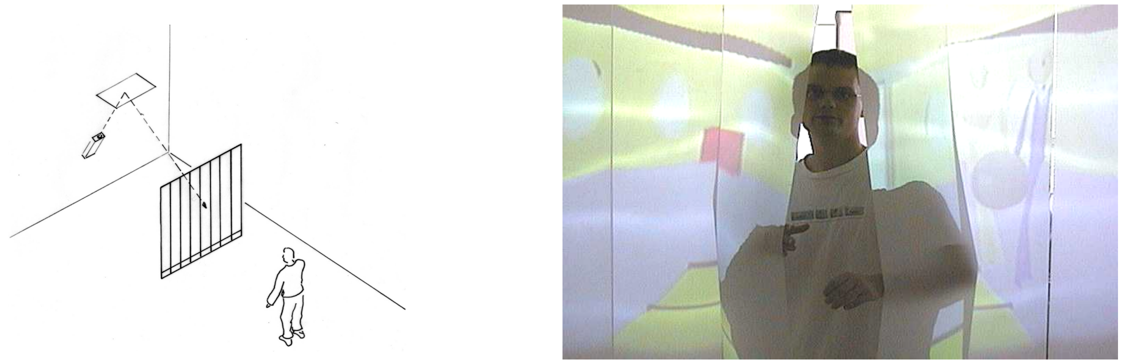
This chapter has developed the idea of traversable interfaces that give the illusion that participants in a local physical environment can completely cross into a remote virtual (or indeed physical) environment and vice versa. The key innovation in the chapter is the extension of the now familiar illusion of entering a remote environment to include appearing to leave one's current environment. We have argued that this is particularly important when the interaction may be observed by people in the two environments as well as experienced directly by the participants. This will be the case in many performance and entertainment applications, but will also be relevant whenever virtual environments and tele-presence technologies are deployed in shared environments, be they public, working or domestic.

We have presented a general design for a traversable interface between a physical and a virtual environment that combines three key components. The first is the use of Physical Roving Proxies (PRoPs) to allow a virtual participant to enter a physical environment. The second is the use of VR technologies to allow a physical participant to enter a virtual environment. The third is the use of non-solid projection surfaces to allow a participant to seemingly step into a projected image of a remote environment. We have presented four early designs for non-solid projection surfaces, a fabric curtain, a water curtain, a sliding door and a flip-up screen. Finally, we have described a demonstrator that shows one possible realisation of our design.

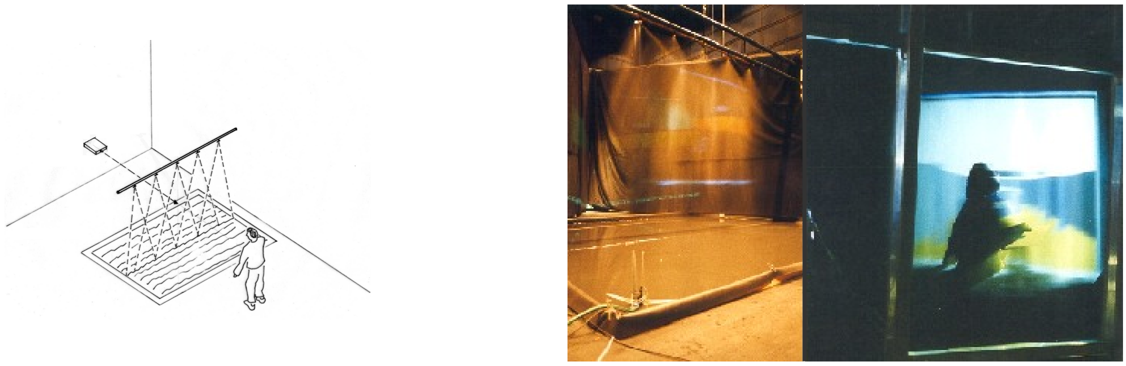
Among the most obvious applications of traversable interfaces are entertainment applications where it may be important to establish a strong illusion of entering a virtual environment. VR-based theme park rides that wish to smooth the transition between watching the ride while waiting for a turn and entering the ride as a participant are a particularly strong candidate, especially as such rides already use animatronic figures and participants occasionally get wet! We also anticipate that our design might be incorporated into more general immersive interfaces. For example, a traversable interface based on our sliding door design might form one side of a CAVE facility, allowing passage to and from the CAVE and providing an external public view of the activity inside.

Our future plans involve developing and evaluating real applications of traversable interfaces. Evaluation will employ ethnographic techniques of the kind that have been previously used to study social interaction in collaborative virtual environments (e.g., Bowers et al., 1996).

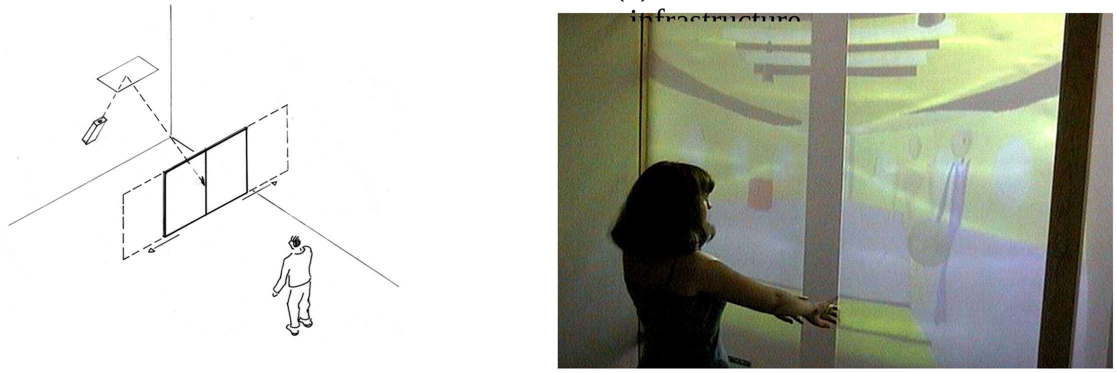
We would like to finish by reinforcing two points that have more general relevance to human-computer interaction. First, is the idea that shared and public interfaces need to be designed with third party observers in mind as well as direct participants. Second, is the observation that virtual reality and telepresence technologies have always been concerned with creating an illusion – the illusion of entering a new and remote environment. This chapter has explored how more traditional theatrical effects, such as moving screens and curtains, and changes in lighting might enhance this illusion, an approach that might be applied to the design of a wide range of human-computer interfaces including, but not confined to, those involved in the participatory events which eRENA seeks to support.



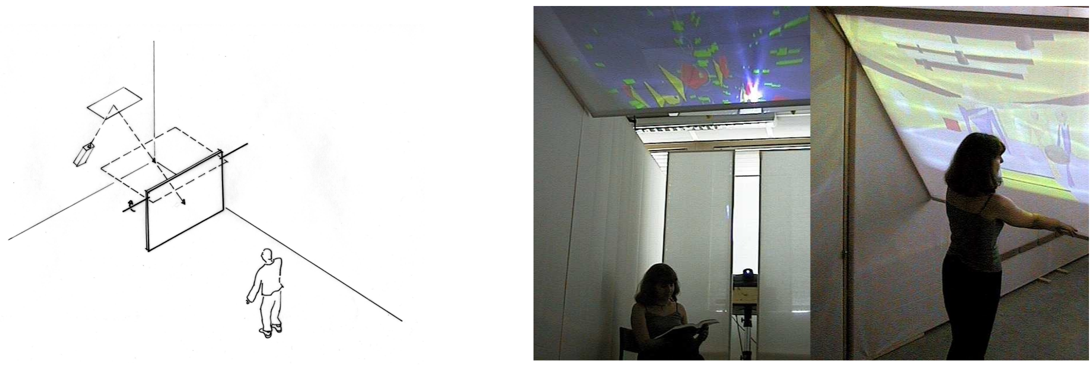
(a) Fabric curtain (b) Emerging through the fabric curtain



(c) Rain curtain (d) Rain curtain infrastructure (e) Rain curtain in use



(f) Sliding door (g) Opening the sliding door



(h) Flip-up screen design (i) Raised as ambient (j) Raising the screen

Figure 4: four designs for non-solid projection

1.6. References

Benford, S. D., Greenhalgh, C. M. and Lloyd, D, Crowded Collaborative Virtual Environments, *Proc. CHI'97*, pp. 59-66, 1997

Benford, S., Greenhalgh, C., Reynard, G., Brown, C., and Koleva, B., Understanding and Constructing Shared Spaces with Mixed Reality Boundaries, ACM ToCHI, 5 (3), 185-223, ACM Press.

Billinghurst, M., Weghorst, S., Furness III, T., Shared Space: An Augmented Reality Interface for Computer Supported Collaborative Work, Proc. CVE'96, Nottingham University, UK, Sept 1996.

Bowers, J., Pycock, J. and O'Brien, J., Talk and Embodiment in Collaborative Virtual Environments, Proc. CHI'96, 1996.

Buxton, B., Living in Augmented Reality: Ubiquitous Media and Reactive Environments, *Proc. Imagina'95*. Monte Carlo, February 1995.

Cruz-Neira, C., Sandin, D. J., DeFant, T. A., Kenyon, R. V. and Hart, J. C., The Cave - Audio Visual Experience Virtual Environment, CACM, 35 (6), pp 65-72, 1992.

Ishii, H. and Ullmer, B, Tangible bits: Towards seemless interfaces between People, Bits and Atoms, Proc. CHI'97, pp 234-241, ACM Press, 1997.

Koleva, B., Benford, S. and Greenhalgh, C, The Properties of Mixed Reality Boundaries, Proc. ECSCW'99, Copenhagen, September 1999, Kluwer.

Kuzuoka, H., Ishimoda, G. and Nishimura, T., Can the GestureCam be a surrogate?, Proc. ECSCW'95, Stockholm, Sweden, September 1995, Kluwer.

Milgram, P. and Kishino, F., A Taxonomy of Mixed Reality Visual Displays, IEICE Transactions on Information Systems, Vol E77-D (12), Dec. 1994.

Paulos, E. and Canny, J., PRop: Personal Roving Presence, Proc. CHI'98, pp 296-303, 1998.

Usoh, M., Arthur, K., Whitton, M., Bastos, R., Steed, A., Slater, M. and Brooks, F., Walking > Walking-in-Place > Flying, in Virtual Environments, *Proc. SIGGRAPH'99*, pp359-364.

Chapter Two
Interaction Surfaces for Participation in Mixed Realities

John Bowers and Sten-Olof Hellström
Royal Institute of Technology (KTH), Stockholm, Sweden

Michael Hoch
ZKM, Karlsruhe, Germany

2.1. Introduction: Interaction Surfaces

In this chapter, we describe some approaches to the use of *interaction surfaces* which have been developed by researchers working in collaboration at KTH and ZKM in the eRENA project. Our interest in interaction surfaces complements the work reported in Chapter 1 concerning boundaries between real and virtual environments. A boundary between two three dimensional environments will tend to be (more or less) two dimensional, that is, a surface. The current work explores the possibility that such surfaces can provide loci for interaction. That is, gesture upon or in relation to the surface has consequences for the relationship between the real and the virtual in an electronic arena or has other interactional consequences. In short, we feel that there is something idiomatic about exploring interaction surfaces in electronic arenas. An interaction surface enables spatialised gesture or the disposition of objects upon it, such forms of spatial interaction being appropriate to environments whose spatiality is essential to the design and experience of them. Accordingly, we have focussed on supporting spatial-gestural interaction with respect to interaction surfaces, rather than, say, more punctate actions akin to switching or button pressing.

We confine ourselves here to flat, bounded, two dimensional interaction surfaces. This is a limit case, for surfaces can be imagined which have a degree of thickness and contour or which wrap-around upon themselves or are of more complex geometrical forms, yet still are surfaces. Our interaction design task is to consider principles by means of which such surfaces can become imbued with rich interactional significance in spite of their simplicity.

One approach is to enable users to interact at the surface with a tool which transduces data on the basis of allowing many degrees of freedom (DOFs) in the gestures employed to manipulate the tool. For example, the currently available Intuos graphics tablets from Wacom (see http://www.wacom.org) use a stylus which allows 9 DOFs of movement (X contact position, Y contact position, tip-pressure, three stylus translational DOFs, three stylus rotational DOFs) to be transduced into 9 data streams which can be made available for applications. Applications of such devices is not confined to graphics: MAX objects exist, for example, to make data from Wacom tablets usable in the MAX/MSP music processing environment. Wacom also market a pressure sensitive LCD screen which enables stylus-based interaction direct on a computer graphics projection.

In our work, we too have explored the use of ‘tools’ and systematically placing them and manipulating them on the surface of a projection as a means for supporting interaction. This is the approach of the RoundTable. A table-top back-projection screen enables images to be displayed. Objects can be placed on top of the table, their positions being tracked by an infra-red sensitive camera mounted above working in tandem with computer vision software. This is a more open platform for experimentation than commercially available solutions. The vision system can be configured to recognise and track multiple objects of varying kinds, each of which can have different consequences for the display or for other aspects of the electronic arena the RoundTable is part of. We describe the physical construction of the RoundTable in the latter parts of this document. To avoid repeating material, the reader is referred to Deliverable D4.5 to find descriptions of how we have developed applications for the RoundTable and the principles by means of which we take data from the vision system and transform it into usable control data. The reader is also referred to Deliverable D4.3/4.4 which contains in Chapter 5 an extensive comparative review of the Round Table with other approaches to physical interaction and so-called ‘tangible bits’. Our presentation of the RoundTable in this chapter concentrates on the details of its physical construction and of the design of the vision system, RT, which it employs.

As an alternative to the manipulation of tools at the interaction surface or the placement of ‘tokens’ upon it, one can take a user’s touching of the surface as the source of peripheral data. Although we are concerned with 2D surfaces, this does not mean that one necessarily only obtains two streams (X and Y position) of peripheral interaction data from a user’s touch. A third dimension (pressure) can be extracted from a pressure sensitive device. Other means of gestural transduction may complement the extraction of data from touch at the surface itself (e.g. video analysis of the trajectory of a hand’s approach). A surface may be designed with multiple interwoven sensors and hence able to report on multiple points of contact simultanously. A user may also wear a pressure sensitive glove, which will permit yet more streams of data to be extracted, and gestures differentiated. In Deliverable D6.1, Part III, we describe some elementary gestural techniques developed at KTH for real-time interaction. Using just a single glove in relation to an interaction surface containing a pressure sensor, we describe techniques for differentiating between ‘poke’, ‘push’, ‘punch’ and ‘fist’ gestures depending on which sensors are indicating above threshold activity. Having ‘dead’ regions on the interaction surface also enabled us to distinguish between gestures which were made in contact with the pressure sensor and those which were not. Our intention was to demonstrate that considerable expressivity can be derived from quite simple peripheral data capture methods in relationship to low dimensional interaction surfaces. When a user can make contact at multiple loci in various combinations on a regionalised interaction surface, it should be clear that the different combinations of data which are distinguishable can be very large.

Another technique for adding to the expressivity that is possible with simple peripheral transduction technologies is to take the temporal sequence with which gestures are performed as having a special significance. For example, an application can be programmed to behave quite differently depending on whether activity is first initiated in region A of the surface or in region B (compare with the experiments reported in Deliverable D6.1, Part III, with a pair of proximity sensors).

In short, there is no reason to suppose that simple low dimensional and low DOF devices and surfaces are impoverished as interaction devices. Used artfully, considerable expressivity is possible and distinctions between many gesture-types can be supported. This is important to emphasise because a common trajectory in interaction research in VR and allied research areas is towards capturing very many DOFs of movement data and engaging in data-intensive and computationally complex techniques of gesture recognition and/or three dimensional whole-body tracking. In eRENA, work of this sort has been conducted (see, especially, the account of *Traces* given in Deliverable D6.5). Our emphasis here is to complement such techniques with those involving simpler devices and peripheral technologies but with careful mapping of the peripherally captured data to application control data. In Deliverable D4.5, we present a generic toolkit for enabling such mappings. Here, we describe some specific applications which demonstrate a number of important interaction design principles which such mapping toolkits can help realise.

The rest of this chapter is organised as folllows. In the next section we describe some of the background to some specific sound control applications which have been developed at KTH in the context of supporting the improvisation of electroacoustic music. As elsewhere in eRENA research, we emphasise the importance of studying applications where improvised activity is to be supported – improvisation involves the greatest challenges for real-time techniques. The applications we have developed make use of real-time data from a simple 2DOF touchpad but transform the data in interesting ways so that highly complex methods of sound synthesis can be engaged with by users – users who do not have to be highly trained musicians to gain something from the interaction experience. This is followed by a description of details of the RoundTable interaction surface which are not found elsewhere in other eRENA deliverables.

2.2. Interaction Surfaces for Engagement with Sound Synthesis in Improvised Electroacoustic Music Performance

In this section, we discuss some techniques which map user-input to control data for real-time interaction with complex synthesised sound in the live performance of improvised electroacoustic music. Our interest in this application area exists because we believe it to be a 'tough case' for interaction technology and design principles for electronic arenas. Current synthesis methods (e.g. physical modeling, see Roads, 1996, for an introductory review) implement complex sound models with many parameters being potentially controllable in real-time. How is this to be made manageable for performers? In non-improvised settings, a 'score' may constrain the possibility-space but improvisation raises our question in full effect.

In music which uses sampled sound or imitative synthesis of existing instruments, established performance practice and instrument design can often enable a small set of control parameters to be identified. For example, many instruments are built with pitch control ready-to-hand so that musics which make pitch variation a main structural means can be easily played. In contrast, electroacoustic music often uses forms based on 'spectro-morphological' variation. That is, the dynamic change of the entire acoustic spectrum of sounds provides musical interest. This requires means for organising 'music in the continuum' rather than 'music on the pitch-lattice' to use the terminology of Wishart (1997). Frequency-parameters in a sound model will then be just one set among many with interactive potential. Improvised electroacoustic music, in these respects, has more the character of exploring a multi-dimensional 'soundscape' than varying series of pitches.

While it is now possible to synthesise very rich sounds in real-time with affordable technology, rather less work has been done on formulating novel interaction design principles which would enable new instruments or performance practices to be systematically developed. There are design challenges both for peripheral devices and for uncovering principles for the transformation of data from such devices into an interactionally useful form. The case of improvised electroacoustic music performance is an important application area because it requires techniques which are real-time, 'in the continuum', and enable navigation within multidimensional parameter spaces in ways which are still aesthetically satisfying and suited for public performance. We believe these requirements, if properly attended to, would not merely enhance participants' abilities to creatively engage with sound in an electronic arena but would also provide creative input to research in human-computer interaction on auditory interfaces (for an introduction, see Kramer, 1994), while enhancing interaction design perspectives in computer music (Roads, 1996).

2.2.1. Interaction Design Principles

Our performance interfaces have been built guided by a set of interaction design concepts, which, reciprocally, the interfaces are intended to demonstrate.

Algorithmically mediated interaction. At various junctures in eRENA (see Deliverables D2.2, D6.1, Part III, and D4.3/4.4, Chapter 6) we claim that alternatives to the direct manipulation (DM) paradigm are often required for interaction with complex higher dimensional systems in creative settings such as electronic arenas. Particularly, we argue that close attention should be given to the algorithms which enable captured input (e.g. sensor data) to be used by specific applications. In our work, we separate out a layer of 'algorithmic mediation' so different peripheral devices, transformation algorithms, and sound models can be freely exchanged (cf. the arguments for the MTK in Deliverables D4.3/4.4 and D4.5).

Expressive latitude. We prefer to work with input devices with a small number of DOFs—typically 2D touchpads or small sensor assemblies. We also tend to use devices which are triggered by contact and are not continually coupled to the body. This enables performers to add emphasis to those gestures they make which are actually transduced. Space is left free for expressive body movements which are not sensed and have no technically-mediated musical outcome. The cost of allowing such 'expressive latitude' (see Deliverable D2.3, Chapter 4) is that fewer input data streams are available. We try to compensate for this by careful design of the algorithmic layer.

Third party legibility. In artistic and entertainment performance settings, interaction with technology is a public phenomenon (see Deliverable D2.3, Chapter 4). That is, the engagement that a user has with her technology is also often visible to other participants. Naturally, this is the essence of performance. A performer's gestures with respect to their instrument, prop or interface are done in such a way that they can be understood by others. Indeed, this is another reason for allowing what we have called 'expressive latitude' and preferring interaction surfaces over higher DOF devices. The more space that is left free, the more performers can emphasise what they are doing without fear that this will be needlessly and misleading transduced. In common with the work in Chapter 1, then, we are concerned to design surfaces that allow 'third parties' (e.g. audience members or co-performers) the opportunity to apprehend what's going on in interaction.

Dynamic adaptive interfaces. The algorithms we have designed for interaction surfaces often make for interfaces whose relation to sound dynamically changes. Input may be rescaled in ways that change over time and in response to ongoing user activity. Thus, the interface may 'map' one region in the possibility-space of the sound model at one moment, and a different region later on. In a sense, then, we use low DOF devices to cut a series of sections through higher dimensional spaces over time. Making a simple interface adaptive is another way to enhance its expressivity.

Anisotropic interaction spaces. Direct use or linear rescalings of input data to control parameters creates what can be called an isotropic interaction space. For example, a touchpad which linearly maps pitch to the X dimension and loudness to Y (a common technique in commercial synthesisers and processing units which use touchpads or joysticks) would create a space where movement in a given direction would tend to have the same consequence, e.g., within limits, moving up would always make things louder. In contrast, we use non-linear and discontinuous mappings to create anisotropic interaction spaces. The significance of movement then becomes context-dependent and locales can emerge with different interactive characters.

2.2.2. Two Demonstrators: *Geosonos* and *SO2*

A sensor pad provides XY touch location data at 7-bit resolution to the *Geosonos* algorithm. Several synthesis parameters are jointly controlled by each of the X and Y dimensions but two sets of transformations occur to make the device surface dynamic and anisotropic. First, a histogram counting visits to each of the 128 coordinates is kept for each dimension. These counts are adjusted by a power function which either exaggerates the occurrence of commonly visited coordinates or diminishes or inverts it. A cumulative frequency graph is made of the adjusted counts and used to transform input coordinates. In effect, this rescales the interaction space to stretch or contract it around sounds which have been commonly explored so far. The degree of distortion (and hence the rate the space changes character) is given by the exponent to the power function. In this way, we intend an interface which not merely supports exploring a soundscape but incites it.

Rescaled coordinate values (call them X' and Y') are then further transformed to give output control values for sound synthesis. While several synthesis parameters are jointly derived from X' (or Y'), a different function is used to determine each of them. For example, p1 may be a linear function of X', p2 a sinusoidal function, p3 a quartic, and p4 a discontinuous ramp. This will mean that over the range of the X dimension, the superimposed synthesis parameters will vary in their correlation (sometimes increasing together, sometimes in contrary motion). The overall effect of this is to create a textured interaction surface with regions of variable sonic character. The superimposition of several parameters onto a dynamic 2D surface with different functions applied to each is intended to suggest a geological metaphor. Under the surface, layers slide over and interact with each other: hence *Geosonos*.

SO2 also uses 2D input data but in a different way. A 'generator' is associated with each controllable parameter in the sound model. Generators output streams of values and are specified by setting such details as update rate, range and step size. Random walks and chaotic iterative functions have been used most commonly. A vector of values can be defined specifying the dynamical behaviour of all the generators. The user pre-selects four such vectors as being of interest. These are deemed to be 'virtually' located on the sensor pad such that touching the pad yields interpolated behaviour as a function of the proximity of the touch to the vectors' locations. This also is intended as enriching a featureless 2D surface with a phenomenological sense of a varied stock of sounds present to the touch. Depending on how the interpolation is done and how the generators are defined, changes in behaviour can be smooth or abrupt, again yielding regions with varying character and stability.

2.2.3. Evaluating Our Current Experience

Geosonos and *SO2* have been tested in a number of improvised performances by two groups of electroacoustic musicians: a duo called *Critical* and a trio called *The Zapruda Trio.* In addition, our interfaces have been explored by non-musicians in a number of public demonstrations at our lab and at the CHI2000 conference at The Hague, Netherlands. Both music groups have used our interfaces in public performances and based entire pieces around their use. *Critical* performed at the i3 Conference in 1999 in Siena, at a concert at Fylkingen in Stockholm also in 1999, and, in 2000, at The Royal Music Academy's Research Conference at Huddersfield, UK and as invited performers at Sonic Arts 11, University of East Anglia, Norwich, UK. Both members of *Critical* used exclusively our interaction surfaces for large portions of these performances. *The Zapruda Trio* have also adopted our techniques and used them in two London concerts in 2000. Both groups report considerable satisfaction with the techniques we have devised and success in their use for the organisation of complex sound in improvised settings.

Our techniques seem to be effective in allowing performers to find usable sounds within complex models and make musically interesting spectro-morphological transformations between them through touching and stroking simple sensor pads. With both algorithms, immediately repeated gestures in the same place give similar results but with enough variety to suggest ways of developing the music. Neither approach allows for fine control of sound but that has not been our intention (though combinations with interfaces more attuned to fine manipulation are certainly feasible).

There are limits to our approach as currently demonstrated. In *Geosonos*, several control parameters are superimposed per dimension. While each will be differently mapped, there are limits on how much can be usefully co-varied in this way (about 4 parameters per DOF with our sound models). Also, the mere superimposition of mappings does not guarantee that they will always intelligibly co-vary. Our users' experience, though, is that this is not a decisive objection to our techniques as some region on the interaction surface can usually be found where the co-variation *is* intelligible and interesting sound can be found. We have, nevertheless, added a 'shuffle-assignments' command to *Geosonos,* which enables control parameters to be reassigned to X/Y dimensions and mapping functions to cope with those situations where the entire interaction surface is unusable musically. While this involves a random allocation of parameters to transformations and dimensions, it is adequate practically. Naturally, more predictable results could be obtained by pre-judging which assignments should be made for which sound model.

In a sense *SO2* scales better than *Geosonos* as there can be, in principle, as many 'generators' providing streams of control parameters as is desired. The limit on the number of generators is due to available processing power rather than inherent features of the design concept. However, *SO2* requires preparatory work in defining interesting vectors. In both cases, our techniques would benefit from being tunable in performance so that, e.g., *Geosonos* could find good selections of parameters to map to the pad and candidate vectors for *SO2* could be defined at run-time. Tuning interaction surfaces *in performance time* also seems more consistent with the spirit of music improvisation than does extensive advanced preparation. Certainly, one of our musician-users has been strongly resistant of engaging in excessive preparatory work as "tuning an interaction technology should be no different from tuning a guitar and I don't expect that to take me a day of research!"

An interesting future line of research would involve the use of our touch sensitive surfaces in tandem with visual displays. Currently, our touch pads (being dedicated commercial devices) do not allow the back-projection of images. In our work on the RoundTable (to be introduced shortly) we have extensively worked with images and visualisations projected onto the table. In *Geosonos*, for example, the mapping functions could be visualised by shading an image to show how much they change input values. The superimposition of different mapping functions could lead to an interesting abstract image – an image which would change as the user engages over time with the interaction surface. Such an image might be of aesthetic interest in its own right as well as giving visual cues to where interesting regions might be found in ‘soundspace’. At the moment, the space of sonic possibility available to the user can only be apprehended through playing the sounds themselves and guessing what so-far-untouched regions might sound like. Giving a visual display, then, might help the user avoid less interesting regions.

From time to time musicians have experimented with non-traditional means for sound control as well as extensions to familiar instruments (Roads, 1996). The specificity of our approach is to use low DOF input devices such as interaction surfaces but to algorithmically 'magnify' them to support engagement with complex sound. Here, we focused on two examples of such an algorithmic layer. We believe this makes for instruments which are equally expressive for performance purposes and potentially more accessible by non-virtuosi than higher DOF devices. While we do not support the fine control that DM interface technologies are often celebrated for, in our application area, the improvisation of synthesised music in an electronic arena, precision in this sense is not necessarily what you want. We prefer to support *’usability at the edge of control’*. For applications in an electronic arena where this is the desired interaction experience, our techniques seem appropriate.

2.3. The RoundTable: An Interaction Surface for Mixed Realities

The remainder of this chapter describes the basic principles, the physical construction and the software system of the RoundTable – a mixed reality interaction surface which has been extensively used in eRENA (see Deliverable D4.3/4.4 and D4.5). While the techniques described in Section 2.2 were focused on enriching a simple touchpad with interesting algorithmic transformations of peripheral data, the RoundTable is primarily designed to enrich user-experience through combining visual displays with physical-tangible interaction. In our most recent applications of the Round Table, where we have explored sound control, this interaction surface comes to have a key role in presenting its users with a media-rich sensory environment.

2.3.1. Introduction

The RoundTable has a projection screen in the middle which is used to display graphics, like a map of an electronic arena or the interface for a sound mixing application. The image on the table-screen is rear-projected—that is, projected from underneath the table using a projector and a mirror. The projection screen is approximately 80cm across with a table height of approximately 95cm. For the camera control application described in Deliverable D4.5, physical objects are placed upon the table-top projection screen to deploy cameras, select cameras for transmission (TX in broadcasting terminology), and enable zooming of the display. On a second projection screen next to the table (for example to the left in Figure 2.1), a 3D rendered scene can be displayed from the perspective of the deployed camera. Alternatively, the camera view, as well as the TX view, can be shown on additional monitors in the room-sized environment (e.g. on the monitors to the right in Figure 2.1).

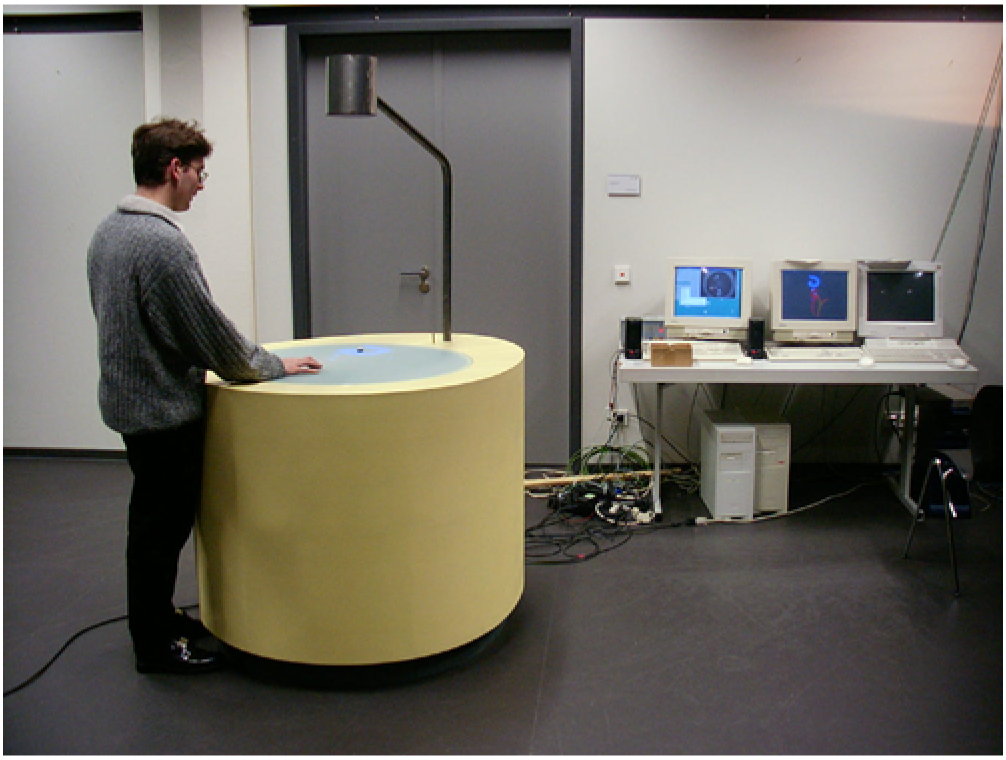


Figure 2.1: RoundTable environment with additional monitors

A pole mounted on the table holds a real camera with infrared light. It is used for tracking blocks that can be placed on the table screen. In our camera control application some of these signify the position of virtual cameras through their location on a representation of the virtual scene. They can also be used in a more generic way in other applications.

In the following we will first describe the motivation for the RoundTable as an input device in terms of general human-computer interaction questions. We will follow this with a description of the hardware setup of the table. Thereafter, the software, RT, for tracking the interaction blocks will be described. Finally, we will outline some of our experiences with the RoundTable before pointing the reader to other sources where specific applications and use-experience is discussed more fully.

2.3.2. The RoundTable as an Input Device

Several usability issues have prompted the RoundTable solution, including the following.

1. Interaction using conventional desktop input devices such as mice, joysticks and keyboards is often too slow when time-critical selections are required (e.g. precisely timed deployments of cameras to where the action is at the very moment it is occurring). To move icons or make selections with a mouse requires the user to first grasp the mouse, then make a controlled movement on screen to the target (icon or menu), engage with the target, and then execute the appropriate function. It is reasonable to believe—and is often claimed—

that with an appropriately designed physical interface, engaging with the target (phicon or push-button) can be accomplished with less preparatory movement.

2. Activities in electronic arenas are commonly co-operative, where multiple users (directors, camera operators and other production members) need to sustain awareness of each other's gestures around shared artifacts—such forms of mutual awareness being very commonly documented as an essential feature of cooperative work in time-critical settings (see, e.g., Martin, Bowers and Wastell, 1997, and the results of the field study of *OOTW* presented in Deliverable D7.a1). This would tend to speak against environments where each participant would only have access to events in an electronic arena through their own monitor display. Environments where views of an electronic arena can be shared and where mutual access to each other's activity with respect to these views can be naturally picked up would seem to be worth exploring. This argument also holds if the RoundTable were being used by co-participating audience members (see our earlier discussions of design for third party legibility).
3. Real-world space needs to be recognized and reserved for participants to bring freely whatever real-world documents and other artifacts they wish allowing interaction with these to be interleaved with technically-mediated interaction with production support applications or virtual environment exploration. Field study in both inhabited television and media art settings in eRENA has revealed the obduracy of paper notes, running orders, and various bits and pieces of equipment which need to be worked with alongside any production software one might wish to develop. These phenomena tend to speak against fully immersive solutions or the hope that everything that a production crew would ever need could be rendered on-screen. For all these reasons, we are investigating physical interfaces and artifacts, on a human-scale, to be sited within room-sized environments, as the appropriate way to make our technologies for production support in electronic arenas available to users. We also prefer embedding displays within flat interaction surfaces for the additional simple practical reason that paper documents and other resources can be placed alongside the displays without falling off!

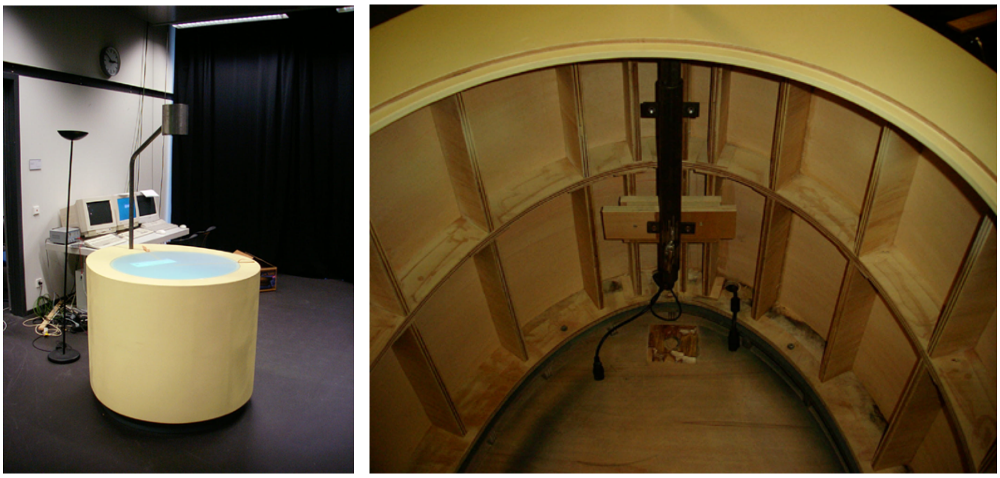
Video as Input Channel

The use of video as the input channel for the RoundTable raises the question of its general applicability and preferability over other solutions. A key point is probably that video is unencumbered. There is no need to put on special things, like markers, or to wear or carry big equipment. Another point is the variety of levels of scale that can be achieved by using video techniques: pointing to rather small objects, like in the camera control application of the RoundTable, is possible just as it is possible to point to larger objects. For example, as reported in Deliverables D6.3 and D7b.4, video analysis techniques are used to follow persons in the EVE dome covering an interaction area of approx. 11x8 meter. Hence, a video solution can either be small scale or large scale with the inherent possibility to scale up. This becomes essential in mixed reality sensor systems. Here, video has the advantage over other devices like touch pads, for example, that in principle the system scales up without the need of more hardware. For example, another person can be added easily (up to CPU limits).

Another consideration is, that video itself is useful for tracking objects, as well as, for using it in the application itself. This leads to two roles of video as an input channel: as content and as an input to the interaction device (for examples of the dual use of video, see the artworks of Michael Saup described in Deliverable D7b.2).

2.3.3. Hardware Setup

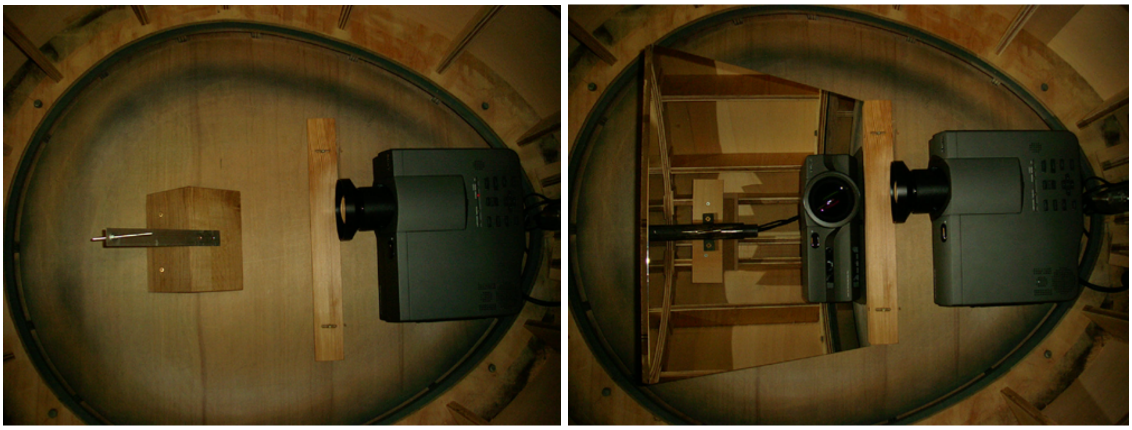
The RoundTable basic construction is made out of wood. It consists out of two pieces: a wooden base with the dimensions of 120x100x15 cm (not clearly visible in Figure 2.2a), and the main cabinet with the dimensions of 130x110x85 cm. The base is slightly smaller in diameter than the main cabinet. Put together the table height is approximately 95cm.



a) b)

Figure 2.2: a) Table exterior construction, b) wooden interior construction view with fixation of the metal pole

On top of the main cabinet, a glass plate with 85 cm diameter fits in so that it gives a smooth surface with the table. We used a standard double glass plate with a milky foil built in-between the two plates. We had to affix an additional matte plastic foil on top of the glass plate to reduce the reflection of the infrared light on the glass. This was necessary to prevent the tracking algorithm from detecting the reflection as a phicon by mistake. Using a glass plate with a sanded surface on the top would also have been possible. A metal pole is used to hold the camera. It is approximately 160 cm high, with a corner at 68 cm and a thickness of 3.5 cm. In the inside a Sony VPL-X600 projector with a 20mm wide angle lens is used to project the graphics on the glass plate. The projector is placed in the inside as shown in Figure 2.3a. A standard foil mirror of size 50x50 cm is placed at approximately 45 degrees as shown in Figure 2.3b. The exact position of the mirror and the tilt can be adjusted.



a) b)

Figure 2.3: a) Table interior with fixation for the mirror, b) same view with mirror and projector in place

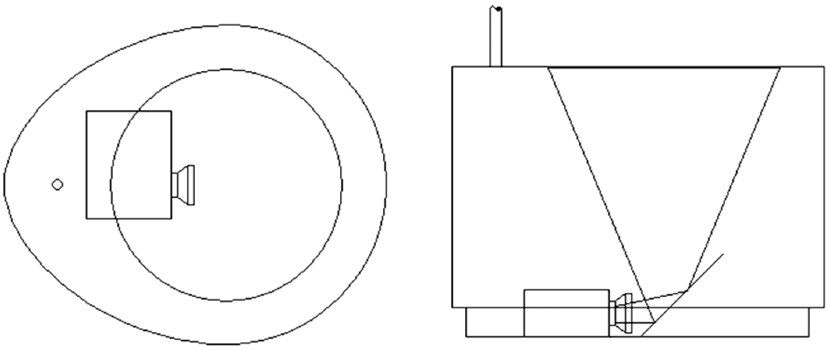


Figure 2.4: Table construction: top view, side view

For the infrared light we use a standard halogen lamp with 60 degrees angle. We use a controlled power source to limit the power consumption to 1.2 amp (12 Volts). This assures that the light source does not produce too much heat, which would damage the infrared filter and/or would make a fan necessary. Also, it further reduces the reflections of the light source from the glass surface. For the camera we chose an infrared sensitive camera (DMK 73mini, 0.05 Lux) with a 60 degrees lens. Hence, if the camera is mounted at a height of approx. 85cm, the glass plate will fit into the video image in the y-direction (in the x-direction the camera sees slightly more).

The table is operated by two PCs: one for the tracking software, one for displaying the graphics on the table. For using the table with additional graphical displays, like a 3D view on a monitor or projection screen, an additional PC or a graphics workstation, like an SGI, would be necessary. Since we use infrared light sources for tracking the objects, we have to make sure that, when setting up the table, no direct sunlight is present in the room and that we use dimmed lab lights. Direct light sources or bright illumination in the room would distract the tracking if the light source contains infra-red light. Since sunlight contains a high percentage of infra-red light the tracking gets distracted even if there is only indirect illumination of the sun in the room.

2.3.4. The RT Software Architecture

In this section, we describe the system architecture of the RT-system. RT stands for ‘Real Time Environment for Tracking Applications’ and comprises the functionality to track things in real space through an abstraction of scenes, (camera) views, objects, primitives, and features. It gives a framework to create tracking applications by offering most of the necessary handling and management routines for real-time tracking. It is, however, not a tracking system itself, i.e. a particular application will be created by adding task specific tracking algorithms. This is achieved by creating and using derived RT-classes and starting the RT-control loop that is described below.

In the following we will first describe the general structure of pattern recognition systems. Thereafter, we will explain the modification and simplifications we did to assure real-time performance.

General Structure of a Pattern Recognition System

The challenge in pattern recognition is to find the best description of the input signal with respect to the current problem space, i.e. an interpretation of segmentation results. In image processing applications the image is, after preprocessing, normally segmented in descriptive parts. To achieve this, a wide variety of segmentation algorithms exist to divide the image into segmentation objects (e.g. into lines or regions). These objects are then matched against a knowledge base that contains possible assumptions of the scenes of the problem space. If we look at the data flow of pattern recognition, we can differentiate the components as shown in Figure 2.5 (Paulus, 1994). The problem of finding the best match to the models and a good segmentation can be seen as an optimization problem. Here, we have to choose between an optimal match and computation time. This becomes specially important with real-time applications.

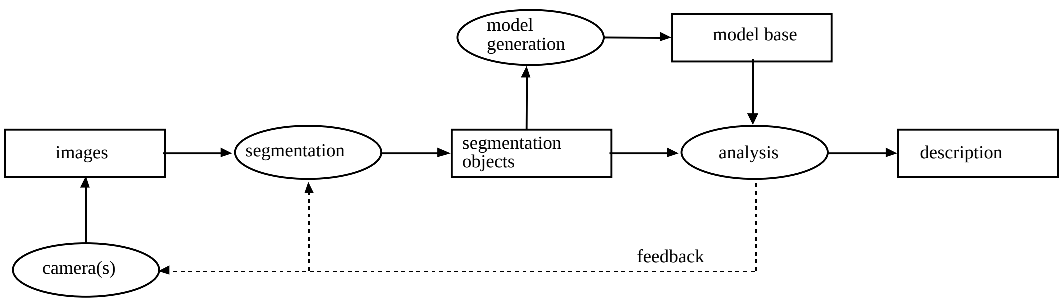


Figure 2.5: Data flow of pattern recognition systems from Paulus (1994)

For knowledge based pattern analysis this simple data flow model is not sufficient. The architecture that a lot of systems follow will have a separate module for *methods*, that comprise the algorithms for preprocessing, feature extraction and segmentation. These methods should in general be problem independent and should not explicitly have a knowledge representation. The application specific knowledge will be held in another separate *knowledge* module. By using this knowledge, yet another *control* module will then determine which methods are used during the processing. Finally, a module called *results database* will store intermediate results during the analysis. The advantage of such a module structure is a general approach to image processing. By using an object-oriented programming language the modules, the data, and the methods can be organized in class hierarchies. This assures the maintainability of such a system. However, the general approach also comes with a more or less large amount of additional administration handling to keep the data structures up to date and for necessary communication between the modules. This will, specially in real-time applications, limit the frame rate that can be achieved.

The RT system follows a somewhat intermediate approach. The focus was more on simplicity and speed than on conformity with a general structure.

RT Data Flow

Deviating from the data flow model shown in Figure 2.5, which describes a data driven approach, we use a model driven approach. The data driven approach tries to instantiate a model by using the segmentation results. The model driven approach tries to find a particular model in the image. The modified data flow is shown in Figure 2.6.

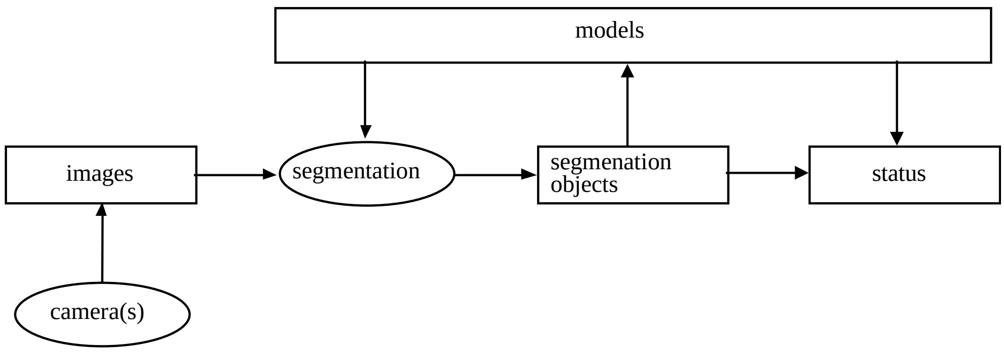


Figure 2.6: model driven data flow

In real-time systems it is often not possible to generate hypothesis from the segmentation results, then, rate them, to, finally, instantiate a model. Often, a hypothesis has to be chosen at some time and, eventually, the process has to start over again. By using the structure as shown in Figure 2.6, where models are able to directly influence the segmentation phase, this step is saved at the cost of a loss in generality of the approach. Here, the pattern analysis task consists of the recognition of the current model state and, thereafter, a tracking with a high frame rate.

Control Structure

In the general case a module called *control* is responsible to choose the appropriate methods by using the knowledge base. The used control structure highly depends on the application. We can differentiate between *interactive* structures with user control, *hierarchical* structures (bottom-up), *model based* structures (top-down), or *mixed* and *data base oriented* approaches (Niemann, 1990). If the sequence of actions is the same for all models, the control structure need only be defined once. In this case a separate control module is not necessary. The RT software, described in this section, has a fixed control structure frame that gets influenced by the models. The models can add additional algorithms to this structure. Hence, besides the knowledge, parts of the control structure of the system are implicitly defined in the models. A separate control structure module does not exist. This results in a disadvantage of not being problem independent, but also in the advantage of a simple and efficient control of the segmentation. By extending the system with new models, new control structures can be added to the system.

The given control structure frame assumes that moving objects are to be tracked in a sequence of images taken by one ore more cameras. The cameras provide views of real scenes. The projected objects consist of a number of primitive elements (e.g. regions). To differentiate between objects of the real world and their projections in a camera image, we call objects of the scene, i.e. real objects, *things*, objects in the image are called *objects*. The primitives, as well as the dependency between the primitives of an object, are defined by the models. A primitive consists of a number of features that need to be segmented in the image. After segmentation, the position of all things in the scene get determined. At the end of the processing loop, a model can predict the position of the primitives in the next frame. In the following, the control structure of RT is shown. If a model based approach is used, the update of the model must be performed in thing->update(). If no model is required for the tracking task, RT supplies a "dummy" model that acts transparently. This results in a common control structure independent of the model used.

```
PROCEDURE testtrack
// instantiate things to be tracked
....

// initialize things (and objects)
FOR ALL rtThing = tracked thing
  rtScene->addObject(rtThing)

WHILE(true)
  FOR ALL rtView = view of scene
    get current image
    // find()
    FOR ALL rtObject = tracked object in image
      FOR ALL rtPrimitive = primitive of rtObject
        rtPrimitive->segment()

    // determine position from object in image
    FOR ALL rtThing = thing of scene
      rtThing->update()

    // predict position of primitives in next frame
    rtThing->model->predict()

  send_data_to_network ()
  ENDWHILE
END


PROCEDURE segment(rtPrimitive)
// preprocessing: e.g. binarize
rtPrimitive->preprocess()

// extract features:
rtFeatures->extract()
FOR ALL rtFeature = extracted feature
  // prevent multiple matches of primitives to objects
  IF CENTER_OF_MASS(rtFeature) already taken
    NEXT
  ELSE
    found = TRUE; BREAK

// set lost state
IF NOT found for more than 4 frames
  lost = TRUE
ELSE
  lost = FALSE
```

```
END
```

Tracking Physical Objects

For tracking the physical objects, the analysis is based on infra-red illumination, luminance-level segmentation, and blob analysis. The image processing system tracks the blocks via a relatively low-cost off-the-shelf infra-red camera set-up that is mounted on the pole of the table. The information that is currently extracted is position data, shape information of all blocks, as well as orientation information for a triangular shaped block. For robust segmentation of the blocks on the projection table, we use retroreflective color attached to each block (made available by 3M, Neuss, Germany) and an infrared filter on the camera to eliminate visible light. For each block present in the scene, this enables the detection of a brighter reflection spot than would be possible with unfiltered room lighting incident upon less reflective surfaces. This greatly facilitates tracking by enhancing the contrast in the image that is input to the analysis routines.

To differentiate the different shapes (see Figure 2.7) we have to find a representation of the form of the object. We calculate simple form parameters that are measurable properties of the detected regions. In the case of a bi-level image it is shape, position, and orientation that conveys meaning, so it is the measurement of these properties that is crucial. For detecting the phicons we will determine the properties area, perimeter, compactness, and roughness:

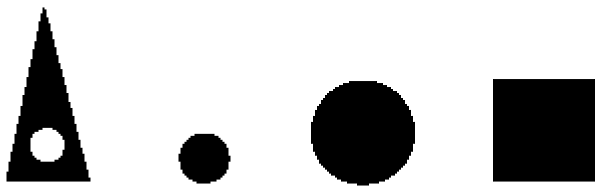


Figure 4: Shapes of detected phicons in our camera deployment application.

area — The area of a region is most simply expressed as the number of pixels comprising that region. The physical area is found by multiplying the number of pixels by the area that was sampled by each pixel. For example, a region containing 10 pixels, each of which represents 1.2 square feet, has a physical area of 10x1.2 = 12 square feet. If the size of a pixel is known, the physical area can be computed, but most often the area is just expressed as the number of pixels. To get a measurable property independent from the image resolution used, we calculate a so called area index by normalizing by the number of pixels in the image (for readability purposes this value is multiplied by 500).

$$area\_index = 500 \bullet blob.area \bullet image.xsize \bullet image.ysize$$

perimeter — Computing the perimeter of a region is more difficult than computing its area. In a bi-level image, the perimeter of a region consists of the set of pixels that belong to the object and that have at least one neighbor that belongs to the background. The perimeter gets calculated by starting at one point at the perimeter and then collecting all neighbor object pixel in a counter clockwise manner. The perimeter is then used to calculate the next parameters. This results in a parameter that counts the total length of edges in a blob with an allowance made for the staircase effect that is produces when diagonal edges are digitized (inside corners are counted as 1.414, rather than 2.0).

compactness The compactness of an extracted blob is calculated by the following formula, whereas A denotes the area, represented by the number of pixels in a blob, and p is the perimeter of the blob. If the blob has the shape of a circle, c equals 1.

$$compactness = perimeter^2 / (4\pi A)$$

roughness for this measure we determine the width and height of the bounding box of a blob at four different angles (0, 45, 115, and 155 degrees) by resampling the image at these angles. We then use the maximum difference between these widths and heights as a factor for determining the roughness. Indeed this measure could also be named circularity since it would give the least value for a perfect circle and continuously increases for off circle shaped objects. The factor of 6 in the following is again a scaling factor. In this case it conforms the calculated parameter range more with the Matrox Image Processing Library (MIL) which we formally used.

$$roughness = 6 \bullet max\_difference / \Sigma(width + height)$$

Camera Selection and Direction

Apart from these features, we use the bounding-box and center-of-mass as properties to track and identify the objects. Furthermore, for the camera phicon the number of holes is determined to differentiate between a selected and a deselected camera. To determine the direction of the triangle phicon, we use three of the four contact points of the corresponding blob (contact points are the pixels that touch the bounding box at x_min, x_max, y_min, and y_max). These points are then ordered and sorted out to get the three points that actually built the triangle. By not using a equal sided triangle, it is easy to determine the direction of the camera.

2.3.5. An Outline of Our Experience with the RoundTable

We have been using the RoundTable for nearly two years in the eRENA project and have designed a number of applications for it. The table and its associated applications have been presented to a large number of people independent of the eRENA project and their assessments have been documented and their use observed. We give detailed accounts of use-experience alongside our descriptions of the applications we have worked with. This appears in Deliverable D4.5. Here, we make some generic remarks about the utility of the RoundTable as an interaction surface.

Since we use infra-red light to track the objects we have encountered problems with interference from sunlight and other light sources as anticipated above. These problems can be overcome when sunlight is blocked from the room and light sources are dimmed. However, it is not always possible to insist on a darkened environment in practical settings. Additionally, embedding the RoundTable in a setting with little extraneous light can make it difficult for users to bring documents and other resources if these need light to be read or themselves emit light. Dissipated light from the projection itself is not always enough to enable other resources in the room to be inspected clearly or used efficiently. It has to be admitted that these consideration place practical limits on the use of video tracking as we currently implement it. Alternatives are possible if ambient light cannot be controlled. For example, a camera can be mounted underneath and/or objects could be recognised through bar-code detection. Embedding wireless transmitters in interaction objects is also a (more costly) possibility. These alternatives too have advantages and disadvantages. Considering the available options, none are perfect – the RoundTable, from this point of view, being no worse than others.

Other problems we encountered concern the way users touch the objects, i.e. when users grasp objects from above they often partly (or fully) occlude the objects from the camera. This can make the tracking erroneous. However, once the user gets instructed how to handle the objects we do not encounter this as a major problem. Indeed, as discussed in Deliverable D4.5, we have noticed users spontaneously discover that covering an interaction object is a gesture that can be creatively used (e.g. to simulate the momentary removal of an object). Again, if such occlusion problems become excessive, it is possible to consider siting the camera beneath the table.

Some object manipulation and detection problems can also be ameliorated by careful phicon design. Remember, one of our phicons is a triangular block with a hole in the middle. We have noticed very occasionally that when users touch the phicon with one finger, to move it a little bit, they open up its hole as far as the tracking system is concerned. This can lead to the object being misrecognised as its compactness can now more closely approximate the figures associated with one of the other phicons. We do not regard such difficulties as being decisive as there is considerable scope for varying the designs, shapes and dimensions of even the simplest geometrical forms as interaction blocks.

Our initial motivation for the physical interaction solution that the RoundTable embodies was based on the claim (often articulated in the literature) that interaction with phicons is commonly fatser than with conventional desktop widgets. This claim is critically interrogated in Deliverable D4.5 where we are able to compare RoundTable with conventional workstation implementations of essentially the same applications. Some operations are noticeably quicker, but others seem slowed. This again enables us to know the limitations of the RoundTable as a solution (see Deliverable D4.5 for details).

It is to be noted that our interaction blocks are not motorised or remote controlled in any way. This means that their positions and orientations cannot be updated without them being manually moved. If one does, by contrast, implement operations which allow the transformation of the visual display, this can lead to situations where the block position is misleading or invalid. This places limits on the sense in which a block can be used to *represent* an object in the graphical environment. Even simple display manipulation functions like zooming or scrolling introduce these discrepancies. Application developers, then, have a choice between *either* allowing such operations and trying to instil a user model which enables users to be tolerant of mismatches *or* disallowing such operations and keeping with a simple application and associated display technique. Our most recent work favours the latter alternative. This also seems most sensible if the RoundTable is designed to support cooperative working. (Full discussion of this also appears in Deliverable D4.5.)

Our overall appraisal of the RoundTable, then, is nuanced. In contrast to much of the HCI literature at the time of writing, we do not think that physical-tangible interaction methods offer categorical advantages over conventional desktop-workstation-based solutions. At least, not so in all contexts. Our experience has enabled us to articulate the boundary conditions on the applicability of such interaction techniques.

2.4. Summary

This chapter has brought together some of the main work in eRENA on interaction surfaces. We have argued for the in-principle utility of interaction surfaces as means for enabling gestural interaction within higher dimensioned electronic arenas. We have suggested that even very simple, low degree of freedom devices can be highly expressive with careful mapping of the transduced data they yield. We have presented a number of applications, together with underlying design principles, to enable interesting capabilities to emerge from simple devices. Our demonstrations in the first half of this chapter further add to the repertoire of techniques we have developed for sound control in eRENA in Year 3. The second half of the chapter outlines the fundamental interaction ideas behind the RoundTable tangible interface that has been used for a number of purposes in eRENA. A technical description of it has been given together with an outline of our experience with it. We conclude that gestural interaction in relation to an interaction surface is an idiomatic way for supporting participants' engagement within an electronic arena. We also hold that our emphasis on surfaces is consistent with the approach to mixed reality boundaries laid out in Chapter 1 and that, together, the two pieces of work clearly indicate how an electronic arena can offer a rich sensorium which is nevertheless sensitive to human needs and abilities.