Human-Computer Interaction Design for Emerging Technologies

Virtual Reality, Augmented Reality and Mobile Computer Systems

Jesper Kjeldskov

Human-Computer Interaction Design for Emerging Technologies

Virtual Reality, Augmented Reality and Mobile Computer Systems

Jesper Kjeldskov

MA, Aalborg University (2000)

Submitted to the Faculty of Engineering and Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at

Aalborg University, Denmark

Human-Computer Interaction Design for Emerging Technologies

Virtual Reality, Augmented Reality and Mobile Computer Systems

Jesper Kjeldskov

Abstract: This thesis deals with interaction design for a class of upcoming computer technologies for human use characterized by being different from traditional desktop computers in their physical appearance and the contexts in which they are used. These are typically referred to as *emerging technologies*. Emerging technologies often imply interaction dissimilar from how computers are usually operated. This challenges the scope and applicability of existing knowledge about human-computer interaction design.

The thesis focuses on three specific technologies: virtual reality, augmented reality and mobile computer systems. For these technologies, five themes are addressed: current focus of research, concepts, interaction styles, methods and tools. These themes inform three research questions, which guide the conducted research.

The thesis consists of five published research papers and a summary. In the summary, current focus of research is addressed from the perspective of research methods and research purpose. Furthermore, the notions of human-computer interaction design and emerging technologies are discussed and two central distinctions are introduced. Firstly, interaction design is divided into two categories with focus on systems and processes respectively. Secondly, the three studied emerging technologies are viewed in relation to immersion into virtual space and mobility in physical space. These distinctions are used to relate the five paper contributions, each addressing one of the three studied technologies with focus on properties of systems or the process of creating them respectively.

Three empirical sources contribute to the results. Experiments with interaction design inform the development of concepts and interaction styles suitable for virtual reality, augmented reality and mobile computer systems. Experiments with designing interaction inform understanding of how methods and tools support design processes for these technologies. Finally, a literature survey informs a review of existing research, and identifies current focus, limitations and opportunities for future research.

The primary results of the thesis are: 1) Current research within human-computer interaction design for the studied emerging technologies focuses on building systems ad-hoc and evaluating them in artificial settings. This limits the generation of cumulative theoretical knowledge. 2) Interaction design for the emerging technologies studied requires the development of new suitable concepts and interaction styles. Suitable concepts describe unique properties and challenges of a technology. Suitable interaction styles respond to these challenges by exploiting the technology's unique properties. 3) Designing interaction for the studied emerging technologies involves new use situations, a distance between development and target platforms and complex programming. Elements of methods exist, which are useful for supporting the design of interaction, but they are fragmented and do not support the process as a whole. The studied tools do not support the design process as a whole either but support aspects of interaction design by bridging the gulf between development and target platforms and providing advanced programming environments.

Menneske-maskine interaktionsdesign for opkommende teknologier

Virtual Reality, Augmented Reality og Mobile Computersystemer

Jesper Kjeldskov

Resume: Denne afhandling handler om interaktionsdesign for en type af nye computer teknologier karakteriserede ved at være forskellige fra traditionelle kontorcomputere i deres fysiske udseende og deres anvendelsessammenhænge Disse bliver ofte refereret til som o*pkommende teknologier*. Opkommende teknologier involverer ofte interaktion som er forskellig fra den måde, hvorpå computere normalt bliver betjent. Dette udfordrer anvendelsespotentialet af eksisterende viden om menneske-maskine interaktionsdesign.

Afhandlingen fokuserer på tre specifikke teknologier: virtual reality, augmented reality og mobile computersystemer. I forhold til disse teknologier adresseres fem temaer: forskningsfokus, begreber, interaktionsformer, metoder og værktøjer. Disse temaer bidrager til formuleringen af tre forskningsspørgsmål, som er styrende for den gennemførte forskning.

Afhandlingen består af fem publicerede videnskabelige artikler og en sammenfatning. I sammenfatningen adresseres det nuværende forskningsfokus udfra forskningsmetoder og forskningsformål. Derudover diskuteres betegnelserne menneske-maskine interaktionsdesign og opkommende teknologier, og to centrale distinktioner introduceres. For det første opdeles interaktionsdesign i to kategorier med fokus på hhv. systemer og processer. For det andet ses de tre teknologier i forhold til tilstedeværelse i virtuelt rum og mobilitet i fysisk rum. Disse distinktioner anvendes til at relatere de fem artikelbidrag, hvor hver af dem adresserer én af de tre studerede teknologier med fokus på hhv. egenskaber ved systemer og udviklingsprocesser.

Tre empiriske kilder bidrager til resultaterne. Eksperimenter med interaktionsdesign bidrager til udviklingen af begreber og interaktionsformer anvendelige for virtual reality, augmented reality og mobile computersystemer. Eksperimenter med design af interaktion bidrager til forståelse af hvordan metoder og værktøjer supporterer designprocesser for disse teknologier. Endeligt bidrager en litteraturundersøgelse til en vurdering af eksisterende forskning og identificerer nuværende fokus, begrænsninger og muligheder for fremtidig forskning.

De primære resultater er 1) nuværende forskning indenfor menneske-maskine interaktionsdesign for de tre studerede opkommende teknologier fokuserer på at udvikle systemer ad-hoc og evaluere dem i kunstige omgivelser. Dette begrænser udviklingen af kumulativ teoretisk viden. 2) Interaktionsdesign for de tre opkommende teknologier kræver udvikling af nye brugbare begreber og interaktionsformer. Brugbare begreber beskriver unikke egenskaber og udfordringer ved en teknologi. Brugbare interaktionsformer modsvarer disse udfordringer ved at udnytte en teknologis unikke egenskaber. 3) Interaktionsdesign for de tre studerede teknologier involverer nye anvendelsessituationer, afstand mellem udviklings og anvendelsesplatform og kompleks programmering. Metodeelementer eksisterer som er brugbare til at støtte interaktionsdesign, men de er fragmenterede og understøtter ikke processen i sin helhed. De studerede værktøjer støtter heller ikke designprocessen i sin helhed men supporterer enkelte dele af interaktionsdesign ved at bygge bro mellem udviklings og anvendelsesplatform og stille avancerede programmeringsfaciliteter til rådighed.

Preface

This thesis deals with human-computer interaction design for three emerging technologies with particular focus on concepts, interaction styles, methods and tools. The thesis consists of this summary and the five individual paper contributions listed below:

1) Kjeldskov J. (2001) *Combining Interaction Techniques and Display Types for Virtual Reality.* In Proceedings of OzCHI 2001, Perth, Australia. Churchlands, Edith Cowan University Press.

2) Kjeldskov J. and Kolbe N. (2003) *Combining Input Devices and Interaction Styles for Handheld Computers.* Extended version of Kjeldskov J. and Kolbe N. (2002) *Interaction Design for Handheld Computers.* In Proceedings of the 5th Asian Pacific Conference on Human-Computer Interaction, APCHI 2002, Beijing, China. Beijing, Science Press.

3) Kjeldskov J. (2003) *Lessons from Being There: Interface Design for Mobile Augmented Reality.* In Andersen P. B. and Qvortrup L. (eds.) Virtual Applications: Applications With Virtual Inhabited 3D Worlds. Berlin, Springer-Verlag.

4) Kjeldskov J. and Stage J. (2003a) *Interaction Styles in Development Tools for Virtual Reality Applications.* In Halskov K. (ed.) Production Methods: Behind the Scenes of Virtual Inhabited 3D Worlds. Berlin, Springer-Verlag.

5) Kjeldskov J. and Stage J. (2003b) *The Process of Developing a Mobile Device for Communication in a Safety-Critical Domain.* In Proceedings of the 9th IFIP TC13 Conference on Human Computer Interaction, Interact 2003, Zürich, Switzerland. Zürich, IOS Press.

During the three years of Ph.D. studies leading to this thesis I have been surrounded by a number of talented people. I would especially like to thank the friends and colleagues I have shared offices with during this period of time: Mikael B. Skov and Hans Olav Omland (Aalborg), Connor Graham, Sonja Pedell and Jeni Paay (Melbourne) for great times, good discussions, valuable guidance and fruitful competition. In addition, I also thank my colleagues at the Information Systems research unit at Aalborg University, including Peter Bøgh Andersen, Peter Axel Nielsen, Palle Nowack, Jeremy Rose, Ivan Aaen, Lars Mathiassen and Helle Schroll. Also I thank colleagues associated with Aalborg University's VR Media Lab during my time of research there, including Erik Granum, Erik Kjems, Peer Møller Ilsøe, Thomas Dorf Nielsen, Ellen Christiansen, Tom Nyvang, Claus A. F. Rosenstand, Moritz Störring, Claus B. Madsen and Jens F. Jensen.

The research presented in this thesis was conducted in collaboration with a number of people. I especially thank my co-authors Jan Stage and Nikolaj Kolbe for inspiring collaboration and good discussions. Also, I thank the editors of the Staging Book Series Vol. 3 and 4, Lars Qvortrup, Peter Bøgh Andersen and Kim Halskov Madsen for constructive feedback on my two book chapters. A special thank goes to Connor Graham for his participation in the classification of research papers in Chapter 2, and to Jeni Paay for final comments and editing of this summary.

Furthermore, I thank my supervisor Jan Stage for inspiring and constructive supervision on research papers and on this summary as well as for valuable guidance on how to pursue a research career. I also thank Jan for encouraging me to apply for a Ph.D. scholarship at the Department of Computer Science at Aalborg University.

During my Ph.D. studies, I spent six months at the Interaction Design Group of the Department of Information Systems at University of Melbourne, Australia. This was a great visit both professionally and personally, and I would like to thank everyone who participated in the "TramMate" project, including Steve Howard, Jennie Carroll, Frank Vetere, John Murphy, Connor Graham, Sonja Pedell, and Jeni Paay. Also I wish to thank everyone at and around the Department of Information Systems, especially Rens Scheepers, Helena Scheepers, Jon Pearce, Michelle Livett, Graeme Shanks, Greg Wadley, Simon Milton, Robert Johnston, and Liz Sonenberg.

Finally, I thank my family for keeping up with me, Rikke Kjeldskov, Søren Bech, Signe Bech Kjeldskov, Maya Kjeldskov, Frey Kjeldskov, Anne Louise Kjeldskov, Hanne Larsen, Ellen Pedersen, Finn Pedersen and Elna Sørensen. Especially, I thank my favorite companion Birte Quist who encouraged me to pursue this and has supported me unconditionally every step of the way.

Jesper Kjeldskov
Aalborg, July 2003

Table of Contents

1 Introduction

This thesis deals with interaction design for a class of upcoming computer technologies for human use characterized by being different from traditional desktop computers in their physical appearance and the contexts in which they are used. Such technologies include for example wearable computers, context-aware computers, immersive virtual spaces and pervasive computerized environments and are typically referred to as *emerging technologies* (Teasley et al. 1996, Cunningham 1998, Gellersen et al. 1999, Perry et al. 2001). Emerging technologies often imply interaction dissimilar from how computers are usually operated. One example of this is location-aware car navigation systems. Such systems are operated by moving through physical space, receiving spoken instructions and pressing a few dedicated buttons within close proximity of the steering wheel and do not apply WIMP-based interaction as we know it from traditional desktop computers. Consequently, such systems challenge the scope of established human-computer interaction styles and concepts and applicability of established methods and tools for their design.

The research presented in this thesis contributes to the production of knowledge about viable design solutions and processes for human-computer interaction by studying three different emerging technologies: *virtual reality, augmented reality* and *mobile computer systems.*

In the following sections, the notion of human-computer interaction design is presented in order to describe the focus of the thesis, and a distinction between interaction design and designing interaction is introduced and discussed. Following this, the notion of emerging technologies is discussed further with emphasis on the challenges for interaction design. These discussions lead to the statement of three research questions.

1.1 Human Computer Interaction

Human-computer interaction (HCI) is “a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them” (ACM SIGCHI 1992:5). This definition is broadly adopted in HCI literature both explicitly (Preece et al. 1994) and implicitly (e.g. Dix et al. 1998, Shneiderman 1998, Newman and Lamming 1995). Decomposing this definition, four specific areas of HCI research emerge, all engaged with interactive computing systems for human use. In the following I briefly outline these and their individual key activities.

1) Design of interactive computing systems “is about developing interactive products that are easy, effective and enjoyable to use” (Preece et al. 2002:2). Within this focus, the primary activity is the creation of new design solutions.

2) Implementation of interactive computing systems focuses on providing “knowledge about the capability of technology and ideas about how this potential can be harnessed” (Preece et al. 1994:38). Within this focus, the primary activity is building applications.

3) Evaluation of interactive computing systems “is concerned with the process of systematically collecting data that informs us about what it is like for a particular user or group to use a product for a particular task in a certain type of environment” (Preece et al. 2002:317). The primary activity within this focus is usability data collection and analysis.

4) The study of surrounding phenomena addresses issues such as "how the introduction of computers will influence work practices" or "how to make sense of what is going on when people communicate with each other or with machines" (Preece et al. 1994:39-41). The primary activity within this focus is the conduction of user studies.

This thesis focuses on human-computer interaction *design*. Conducting user studies and building and evaluating prototypes support this focus, but implementation, evaluation and the study of surrounding phenomena are not studied as such.

1.2 Design

Design of human-computer interaction can be divided into two distinct categories:

1) Interaction design as *properties* of a system.
2) Interaction design as the *process* of creating a system.

The first category is concerned with the exploration of interaction design in its material form: the way users can interact with the computer. The second category is concerned with the exploration of ways by which this material form can be created. This distinction emphasizes the dual meaning of *design* as a *noun* and as a *verb* respectively. Viewing interaction design as a noun describing properties of a system (*a* design), the primary locus of interest is on understanding and describing the specific characteristics of that system giving it its particular qualities and limitations. Viewing interaction design as a verb describing the process of creating a system (design*ing*), focus is drawn towards understanding and describing activities and work practices contributing to this process.

The distinction between interaction design and designing interaction can be identified broadly within mainstream HCI literature (e.g. Preece et al. 1994, Dix et al. 1998, Shneiderman 1998, Newman and Lamming 1995) as well as for emerging technologies (e.g. Bergman 2002), and is discussed in more detail in the following subsections.

Interaction Design as Properties of a System

Research into interaction design as *properties* of a system is concerned with the invention and refinement of user interface concepts and interaction styles for increasing the usability of computer applications. Motivated by the question of what design solutions work well in which situations and why, the purpose of this category of research is to understand interaction design on a conceptual level and contribute to the development of knowledge about design solutions informing the shape of future systems. Key elements of this category of HCI research is, among others, the development of:

1) Abstract *concepts* for interaction such as mental models, direct engagement, mapping, affordances, locus of attention, monotony, designing for action etc. (Norman et al. 1986, Norman 1988, Raskin 2000, Laurel 1992).

2) Specific *interaction styles* such as command language, form filling, direct manipulation, gesture recognition etc. supporting the use of a variety of input and output devices such as keyboards, pointers, graphical displays etc. (Preece et al. 2002, Shneiderman 1998, Preece et al. 1994, Stuart 1996).

Interaction Design as the Process of Creating a System

Complementing interaction design as properties of a system, research into interaction design as a *process* is concerned with the development of means for supporting the practice of creating interactive computer systems. Motivated by the question of how interaction design can be developed in a process that is both structured and creative, the purpose of this category of HCI research is to understand interaction design on a methodological level and to contribute to the generation of increasing knowledge about design processes informing future system development. Key elements of this category of HCI research is, among others, the development of:

1) General *methods* for supporting the process of developing human-computer interfaces such as user-centered design, contextual design, participatory design, design patterns, object-oriented analysis and design etc. (Winograd 1996, Beyer and Holtzblatt 1998, Greenbaum and Kyng 1991, Borchers 2001, Mathiassen et al. 2001).

2) Concrete *tools* for designing and implementing human-computer interfaces in the form of, for example, graphical development tools and advanced programming environments (Conway et al. 2000, Myers et al. 2000, Shneiderman 2000).

1.3 Emerging Technologies

This thesis focuses on three emerging technologies. The term *emerging technologies* is widely used within the area of human-computer interaction (Teasley et al. 1996, Cunningham 1998, Gellersen et al. 1999, Perry et al. 2001). However, no established definition exists. Describing a technology as *emerging* does not in itself say anything about its specific characteristics and properties except that it is *presently up-and-coming.* In that sense, all technologies have at some point in time been emerging and all present emerging technologies will some time in the future cease to fall within this category. Consequently, the classification of a technology as *emerging* is by definition temporary.

The use of the term emerging technologies within computing denotes a range of various characteristics and focal points. Describing emerging technologies as "digital-frontier technologies" demonstrating "practical and speculative interactivity" (Siggraph 2003) focuses on the *properties* of the technology itself and emphasizes the technology being new and inventive and on the edge of what is technologically possible. Describing emerging technologies as "technologies that are not yet widely adopted" (Roberts 2002) focuses on the *use* of the technology and signifies its potential future propagation. Describing emerging technologies as "technologies that could soon transform computing" (Technology Review 2003) or "technologies that are poised to make a dramatic impact on our world" (MIT 2003) focuses on the *significance* and *implications* of the technology and signifies its potential future influence.

While these views are all relevant for describing different aspects of emerging technologies, they are not individually sufficient for a definition. Combining the focus on properties, use, significance and implications of technology and emphasizing my focus on human-computer interaction, I thus define an emerging technology as:

> *A novel computer technology with promising properties, use and significance, for which the implications for human-computer interaction are still unclear.*

The following briefly exemplifies this definition with four technologies.

A number of novel computer technologies with promising properties, use and significance, for which the implications for human-computer interaction are still unclear, are presently emerging. One example of this is the “Burton Amp Jacket” (Burton 2003) that extends an MP3 player with a wearable sound and control system built into the fabric of a jacket (figure 1a). In order to, for example, change tracks and control the volume, the user touches specific areas on his jacket. Another example is the Disney interactive virtual reality theme park ride “Pirates of the Caribbean” (Sehell and Shochet 2001) in which four people steer a physical boat through a virtual environment while defeating virtual enemy pirate ships, forts and monsters (figure 1b). Interacting with this system, the users collectively operate a physical helm and six cannons on deck. A third example is the context-aware car navigation system “BMW Assist” (BMW 2002) which provides the driver with route planning and other information, based on the car’s location, through a mix of text, graphics and speech and automatically makes emergency calls in a crash situation (figure 1c). Interacting with this system, the user drives the car and receives spoken commands. Finally, the Electrolux “Screenfridge” (Electrolux 1999) is an example of a computerized refrigerator with embedded Internet connection etc. facilitating text and video based communication among members of the family, providing news and suggesting recipes based on the food in it (figure 1d). Interacting with this system, the users scan their groceries and talk to a screen.

a) Wearable music system

b) Immersive theme park ride

c) Context-aware navigation system

d) Embedded communication system

Figure 1. Examples of emerging technologies

These four examples each represent one or more directions of emergence within computing. The car navigation system represents *context-awareness* in information systems (see e.g. Shilit et al. 1994, Dey and Abowd 2000) and *multi-modality* in human-computer interaction (see e.g. Oviatt et al. 2002). The theme park ride represents *immersion* into computer generated virtual 3D environments (see e.g. Rheingold 1992, Murray 1997). The wearable music system represents physical *mobility* of computing (see e.g. Kristoffersen and Ljungberg 1999) and the Internet enabled refrigerator represents *embeddedness* of computer technology into everyday products (see e.g. Weiser 1991, Norman 1998).

For the research of this thesis I have focused specifically on interaction design for 1) virtual reality, 2) mobile computer systems and 3) augmented reality. In light of the discussion above, these technologies are studied in relation to *immersion* into virtual space and *mobility* in physical space. Immersion and mobility are discussed below in more detail followed by a description of the three emerging technologies.

1.4 Immersion and Mobility

The concept of immersion addresses the phenomenon of presence in a computer-generated virtual space. The literature suggests two opposing views on this concept. Quantifying the phenomenon, Slater and Wilbur (1997) define immersion as an objective property of a computer system denoting "the extent to which computer displays are capable of delivering an inclusive, extensive, surrounding, and vivid illusion of reality to the senses of a human participant". In contrast to this system-oriented view, Witmer and Singer (1998) define immersion qualitatively as "a psychological state characterized by perceiving oneself to be enveloped by, included in, and interacting with an environment that provides a continuous stream of stimuli and experiences".

In this thesis, I adopt the latter view of immersion as a property of a person. Extending the definition, I add that immersion is not a binary condition but a continuum. Describing the capability of a computer system to create an immersive user sensation, as emphasised in Slater and Wilbur (1997), I use the term immer*sive.*

The concept of mobility can be defined generally as the quality or state of being able to move or be moved quickly from place to place (Oxford Advanced Learner's Dictionary 1995). This definition, of course, applies to both people and systems. Thus, for example, in the Encyclopedia of Computer Science (Chlamtac and Redi 1998) mobility is described as a property of a portable computer *system* with wireless communication facilities. Contrasting this, the discussions on mobile use of information technology by Kristoffersen and Ljungberg (1999) and Dahlberg (2003) describe mobility as a property of a *person*.

In this thesis, I adopt the latter view on personal mobility. However, I chose to narrow my focus to personal mobility involving computer system use while being mobile. This is done to exclude, for example, situations in which mobility is supported by e.g. looking up traveling information in advance of a trip, or through access to centralized information via distributed non-mobile computer systems. I thus propose the definition of mobility as *a property of a person who moves through physical space while using a computer system.* Like immersion, mobility is not a binary condition but a continuum.

Having defined the concepts of immersion and mobility, I will now describe the three emerging technologies studied within these two directions of emergence.

Virtual Reality

Coined in 1985 by Jaron Lanier, the term of virtual reality denotes "the use of computers and human-computer interfaces to create the effect of a three-dimensional world containing interactive objects with a strong sense of three-dimensional presence" (Bryson 1996). Compared to desktop computer systems, virtual reality facilitates a high level of immersion into virtual space. This is supported by means of special display systems covering a large proportion of the user's visual field of view and interaction devices sensing the user's physical movements. Virtual reality allows the user to step into situations, which would be expensive, dangerous or even impossible to experience in the real world such as simulations of complex industrial installations, buildings not yet constructed or visualizations of objects and phenomena invisible to the human eye such as molecules and airflow.

Mobile Computer Systems

Mobile computer systems can be defined as systems running on small, lightweight, easily moveable, wireless and self-contained computing devices with limited input and output capabilities (Mobile HCI 2002). Compared to desktop systems, mobile computer systems facilitate a high level of mobility in physical space. This is supported by technology being so small that it can be held in one's hand, carried in a pocket or worn as a part of one's clothes. Mobile computer systems allow the user to access digital information while moving through physical space. Personal digital assistants provide calendar-like functionality and applications useful when traveling and mobile phones support spoken and written communication. Other mobile computer systems have even more dedicated functionality such as GPS route planners and MP3 players.

Augmented Reality

Augmented reality denotes a computer system that allows the user to see the real world, with virtual objects superimposed, thus supplementing reality rather than replacing it (Azuma 1997, Behringer et al. 1999). Compared to desktop computer systems, augmented reality facilitates the user being simultaneously present in a virtual and a real environment. This effect is typically created by means of special display systems such as semi-transparent head-mounted displays and interaction devices sensing the user's physical movements (Feiner et al. 1999). Augmented reality allows the user to see objects and phenomena otherwise invisible to the human eye such as hidden structures, people in the dark or fictive actors in a computer game embedded into the physical surroundings.

Mobility and immersion are often described as extremes on a linear continuum from the real to the virtual environment (Milgram and Colquhoun 1999, Tamura et al. 1999, Billinghurst et al. 2001). On this continuum, the three emerging technologies studied in this thesis are positioned as illustrated in figure 2 below.

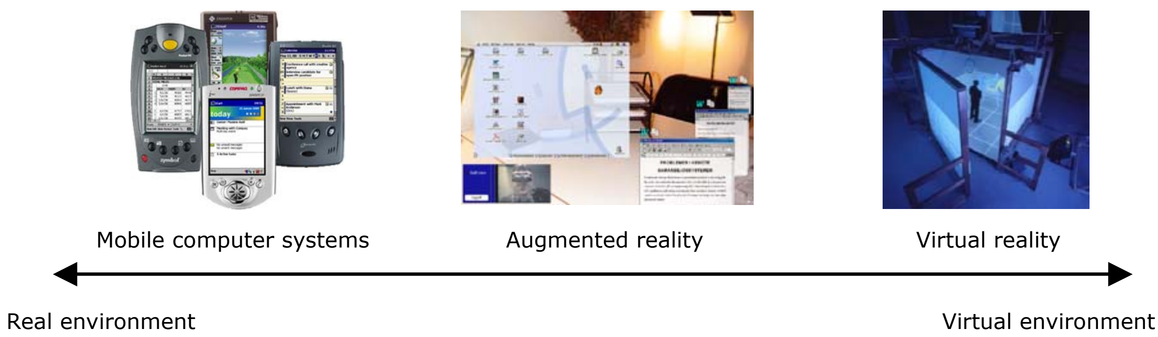


Figure 2. The "reality-virtuality continuum" (Milgram and Colquhoun 1999)

1.5 Research Questions

The discussions of interaction design and emerging technologies leads to the statement of three research questions.

Research Question #1

Human-computer interaction design for the three studied emerging technologies is a growing area of research, but it is unclear what research is currently being done and what the focus is. This leads to the first research question of this thesis:

> *What is the focus of HCI research for the three emerging technologies studied and what are the limitations and the opportunities for future research?*

This research question is addressed through a literature review of HCI research for virtual reality, augmented reality and mobile computer systems. In this review, 220 research papers on the studied technologies are described in relation to their research methods and research purpose.

Research Question #2

From the perspective of interaction design as properties of a system, a key question in the light of emerging technologies is how concepts and interaction styles are challenged by, for example, mobility in physical space and immersion into virtual environments. This leads to the second research question of this thesis:

> *Which concepts and interaction styles are suitable for human-computer interaction design for the three emerging technologies studied?*

This research question is addressed through three papers focusing on concepts and interaction styles for virtual reality, augmented reality and mobile computer systems. In these papers, experimental implementation and evaluation of interaction design informs the generation of new concepts and matrices describing suitable interaction styles.

Research Question #3

From the perspective of interaction design as a process of creating a system, a key question in the light of emerging technologies is how methods and tools can support development for technologies facilitating for example mobility in physical space and immersion into virtual environments. This leads to the third research questions of the thesis:

> *How do methods and tools support the process of designing human-computer interaction for the three emerging technologies studied?*

This research question is addressed through three papers focusing on methods and tools for virtual reality, augmented reality and mobile computer system design processes. In these papers, different design methods and development tools inform the process of designing interactive computer applications.

The first research question will be addressed in Chapter 2. The second and third research questions are addressed in the five paper contributions summarized in Chapter 3.

2 HCI Research for Emerging Technologies

In this chapter I address the first research question by presenting a literature review of human-computer interaction research on emerging technologies providing a snapshot of current research. Inspired by a study within the field of Information Systems (Wynekoop and Conger 1990), this is done by classifying research papers on HCI for the three technologies studied in a two-dimensional matrix, outlining the distribution of research methods and research purpose. On the basis of this, I discuss the way human-computer interaction research for emerging technologies is currently being done.

The chapter is structured in the following way. First, the research methods and research purposes used for classifying research papers are outlined. This is followed by a presentation of the classification procedure and the identified distribution of research. Finally, the findings are discussed in relation to the first research question.

2.1 Research Methods

The first dimension of the HCI research classification describes research methods. Defining and especially differentiating research methods can be a challenge. Definitions are sometimes vague and aspects of different methods can overlap. I have chosen to distinguish between eight research methods extracted from Wynekoop and Conger (1990) with supplementary input from Lewis (1985), Myers (1997), Rapoport (1970) and Yin (1994). In table 1 below, these eight research methods are summarized and grouped in accordance with Benbasat's (1985) categories of *natural, artificial* and *environment independent* settings.

	Method	Strengths	Weaknesses	Use
Natural setting	Case studies	Natural settings Rich data	Time demanding Limited generality	Descriptions, explanations, developing hypothesis
	Field studies	Natural Settings Replicable	Difficult data collection Unknown sample bias	Studying current practice Evaluating new practices
	Action research	First hand experience Applying theory to practice	Time, bias, ethics Unknown generality	Developing and testing hypothesis and theory
Artificial setting	Laboratory experiments	Control of variables Replicable	Limited realism Unknown generality	Controlled experiments Theory/product testing
Environment independent setting	Survey research	Easy, low cost Can reduce sample bias	Context insensitive No variable manipulation	Collecting descriptive data from large samples
	Applied research	The goal is a product which may be evaluated	Produced solutions may be very limited	Product development, testing hypothesis/concepts
	Basic research	No restrictions on solutions Solve new problems	Costly, time demanding May produce no solution	Developing understanding and building theory
	Normative writings	Insight into firsthand experience	Opinions may influence outcome	Descriptions of practice, building frameworks

Table 1. Summary of research methods. Extracted from Wynekoop and Conger (1990)

Case studies are intensive empirical investigations of contemporary phenomena within small size entities such as groups, organizations, individuals, systems or tools in real-life context with the researcher distinct from the phenomena being studied (Yin 1994, Wynekoop and Conger 1990). Case studies are particularly well suited for describing and explaining specific phenomena and for developing hypothesis or theory. However, case studies are very time demanding and generalizing findings can be difficult.

Field studies are characterized by taking place in “the real world” covering a range of qualitative and quantitative approaches from ethnographic studies of phenomena in their social and cultural context to field experiments in which a number of independent variables are manipulated (Lewis 1985, Wynekoop and Conger 1990). The major advantage of field studies is the generation of large amounts of rich and grounded data. Disadvantages include unknown biases and no guarantee of data being representative.

Action research is a method through which researchers not only add to the body of scientific knowledge but also apply that knowledge to the object of interest through intervention and participation in the activity being studied (Myers 1997, Rapoport 1970, Wynekoop and Conger 1990). The advantage of action research is the close relationship between theory and practice. Action research is, however, often very time consuming, and ensuring objectivity and generalizability can be difficult.

Laboratory experiments are characterized by taking place in a controlled environment created for the purpose of research such as in an office (Tang et al. 2001), in a hallway (Bohnenberger et al. 2002), in a simulator (Johnson et al. 1999) or in dedicated laboratories. The major advantage of laboratory studies is its support for focus on specific phenomena of interest with a large degree of experimental control. One disadvantage is the unknown level of generalizability of results to real world situations.

Surveys usually inform research through various systematic techniques such as questionnaires and interviews from a known sample of selected respondents assumed to be independent of their environment (Wynekoop and Conger 1990). The major advantage of surveys is the amount of data gathered, the level of researcher control and the potential generalization of results. On the other hand, surveys suffer from providing only snapshots of studied phenomena and rely on the subjective views of the respondents.

Applied research builds on trial and error on the basis of reasoning through intuition, experience, deduction and induction (Wynekoop and Conger 1990). Typically the desired outcome of an applied research process is known while means of accomplishing it is not. This makes applied research very goal oriented, typically resulting in some kind of product being produced, which can be evaluated against initial goals. The major disadvantage of applied research is that solutions found (if any) may not be generalizable.

Basic research is characterized by trial and error based development of new theories and the study of well-known problems to which neither viable solutions nor methods are known, relying on the competences of the researcher (Wynekoop and Conger 1990). The major advantage of basic research is its openness in terms of approaches and time, allowing a high level of creativity in the search for methods and solutions. However, no guarantee of any outcome being produced exists.

Normative writings cover the body of *non-research* writings about phenomena of interests such as *concept development writings* organizing ideas for stimulating future research, presentation of *truth* describing ideas, which seem intuitively correct, and Benbasat’s (1987) category of *application descriptions* defined as “narratives written by practitioners” (Wynekoop and Conger 1990). The primary advantage of normative writings is that they require little effort to produce compared to theory. Disadvantages include limited theoretical and methodological reflection.

2.2 Research Purpose

Research methods and purpose are often closely related but not necessarily determined by one another. Like Wynekoop and Conger (1990) I thus use the second dimension of the classification of HCI research for emerging technologies to describe research purpose. Populating this dimension I borrow the categories and definitions originally proposed in Basili (1986) and also used in Wynekoop and Conger (1990). These are defined below.

Understanding is the purpose of research focusing on finding the meaning of studied phenomena through e.g. frameworks or theories developed from collected data.

Engineering is defined as the purpose of research focused towards developing new systems or parts of systems such as e.g. an interaction technique for virtual reality.

Re-engineering describes the purpose of research focusing on improving existing systems by redeveloping them such as e.g. adapting a web browser to a small display.

Evaluating is the purpose of research assessing or validating products, theories or methods e.g. the usability of a specific virtual reality application or a theory of interaction.

Describing finally refers to research focusing on defining desirable properties of products e.g. a virtual reality or mobile computer system.

2.3 Classification Procedure

For the literature review, a total of 220 conference and journal papers were classified in relation to their research methods and purpose. These papers constitute all publications on HCI for virtual reality, augmented reality and mobile computer systems between 2000 and 2002 in the 15 conference proceeding series and journals listed in table 2 below.

Conference proceeding series and journals	Virtual Reality	Augmented Reality	Mobile Computers
Conference on Computer-Human Interaction, CHI, ACM	7	2	8
Conference on User Interface Software and Technology, UIST, ACM	0	3	9
Conference on Advanced Visual Interfaces, AVI, ACM	1	0	2
Conference on Computer-Supported Cooperative Work, CSCW, ACM	0	0	2
Conference on Designing Augmented Reality Environments, DARE, ACM	0	10	0
Conference on Virtual Reality Software and Technology, VRST, ACM	20	2	0
Conference on Virtual Reality, IEEE	25	1	0
Conference on Augmented Reality, IEEE	0	14	0
Symposium on HCI with Mobile Devices, Mobile HCI	0	0	30
Symposium on Designing Interactive Systems, DIS, ACM	1	1	3
Symposium on Interactive 3D Graphics, Si3D, ACM	4	0	0
Transactions on Computer-Human Interaction, TOCHI, ACM	0	2	9
Journal of Personal and Ubiquitous Computing, Springer-Verlag	0	3	39
Journal of Virtual Reality, Springer-Verlag	4	4	0
Journal of Computers and Graphics, Elsevier	3	5	6
Total	65	47	108

Table 2. Distribution of classified research papers

For practical reasons, I have chosen to incorporate only publications available through the Internet. Thus, for example, proceedings from the IFIP TC13 International Conference on Human-Computer Interaction (Interact) are not included.

The 220 papers selected for the review were identified by reading through abstracts (and sometimes introductions) of all publications between 2000 and 2002 in the listed conference proceeding series and journals. A paper was selected for the review if it addressed virtual reality, augmented reality or mobile computer systems *and* human-computer interaction. It was omitted if it focused on other technologies or did not involve any aspects of interaction design.

All selected papers were printed, sorted alphabetically and numbered in relation to the technology addressed. The papers were then read one by one and classified in relation to the purpose of the presented work and the research methods applied in achieving this. The classification resulted in a matrix of research methods and purpose populated by paper references for each of the three technologies studied. After having classified all papers, they were scanned quickly to prevent errors and ensure consistency. The 220 papers were then given to a fellow research student, who repeated the procedure. Following this, we compared our classifications by discussing each of the 220 papers one by one in collaboration. This resulted in a final classification (see appendix B).

While in the literature review presented by Wynekoop and Conger (1990) each paper is only attributed to one research method and one purpose, this was not possible with all papers on human-computer interaction. Some of the reviewed papers clearly employed more than one research method and had multiple purposes. A common example of this would be papers presenting a system engineered by applied research and subsequently evaluated in a laboratory. Consequently, such papers were given two entries in the classification.

2.4 Findings

The classification of HCI research papers for the three emerging technologies studied is illustrated in figure 3 below. This figure shows the percentage of references attributed to each of the 8x5 combinations of research methods and research purpose for each of the technologies.

The classification of HCI research papers shows a pattern in the distributions of research methods and research purpose across virtual reality, augmented reality and mobile computer systems.

One of the central observations to be made from figure 3 is that that most combinations of research methods and research purpose are empty or contain only a small number of references. These are contrasted by a few combinations with a large number of references. Another central observation is that most combinations of research methods and research purpose are populated by comparable percentages of references across technologies. These two observations indicate a similarity between research in human-computer interaction design for the three emerging technologies studied and a bias towards particular methods and purposes. The specific findings are summarized and discussed below.

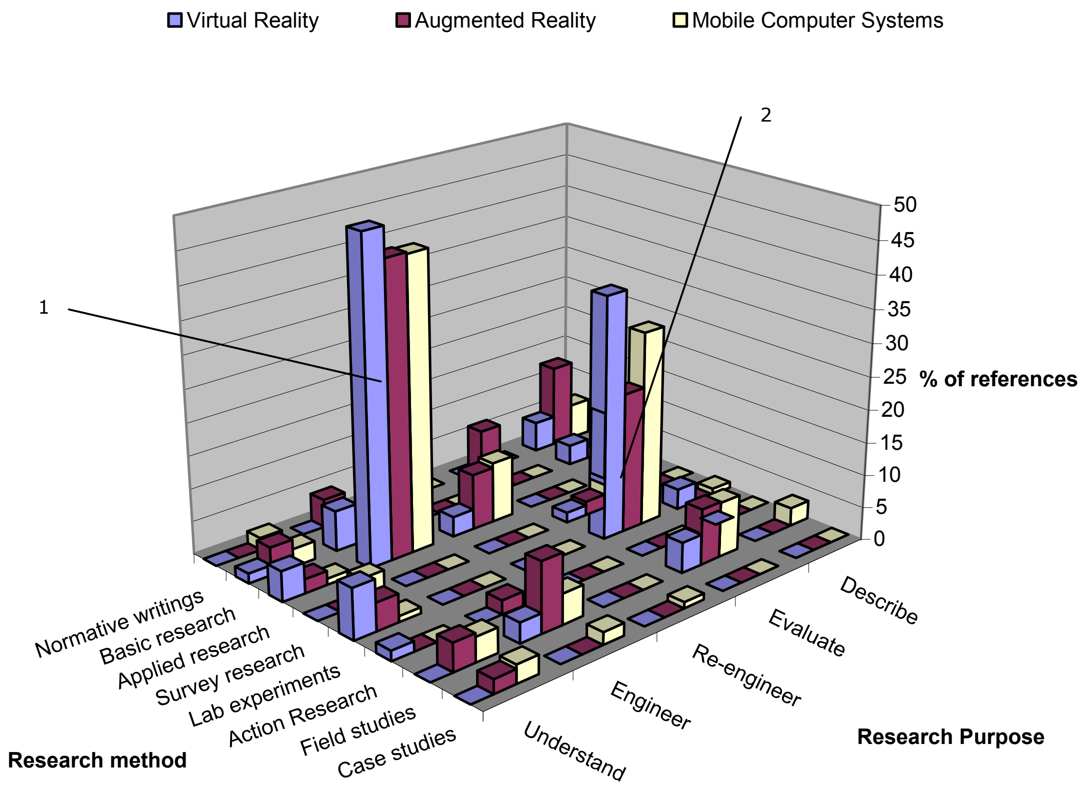


Figure 3. Distribution of human-computer interaction research for emerging technologies based on research publications between 2000-2002 (see appendix B for references and classifications)

The most noticeable groupings of research methods and research purpose are the peaks numbered 1 and 2 in figure 3. Peak number 1 shows that 44-49% of the research papers on virtual reality, augmented reality and mobile computer systems involve applied research for the purpose of engineering. Peak number 2 shows that 21-37% of the research papers involve laboratory experiments for the purpose of evaluation.

The most noticeable gaps in the distribution of research papers are constituted by a general lack of survey research, action research and case studies across purposes. Also, a lack of laboratory experiments for engineering and re-engineering is noticeable.

Looking at research methods, the classification of HCI research for the three emerging technologies studied shows that 61% of the research papers fall into the applied category (135 of 220 papers) representing 68% of the virtual reality papers, 66% of the augmented reality papers and 56% of the mobile computer systems papers respectively. The second most used method is laboratory experiments being applied in 35% of the research (77 of 220 papers) representing 48% of the virtual reality papers, 26% of the augmented reality papers and 32% of the mobile computer systems papers. Next comes field studies, used in 17% of the research reported (37 of 220 papers), representing 8% of the virtual reality papers, 23% of the augmented reality papers and 19% of the mobile computer systems papers respectively. Normative writings form 10% of the research while 6% are basic research and 5% report on survey studies. Only 3% of the research is case studies and less than 1% is action research. This distribution shows a clear bias towards environment independent (applied) and artificial setting (laboratory) research at the expense of natural setting research (case, field and action).

Looking at research purpose, 56% of the research is done for engineering with an additional 8% for re-engineering. Thus in total, 64% of the classified research papers involve building systems (140 of 220 papers). 41% of the papers involve evaluation of systems (91 of 220 papers). Research for the purpose of understanding and describing each appear in 16% of the papers (36 of 220 papers). This distribution shows a clear tendency towards building and evaluating systems at the expense of theorizing and generalizing. Of the 140 papers involving engineering or re-engineering, 84% are based on applied research while only 9% are based on field studies and just 2% on case studies. Thus when building systems for the three emerging technologies studied, there is a clear tendency to do so ad-hoc rather than on the basis of user studies.

In the 91 papers involving evaluation, 73% use laboratory experiments, 18% field studies and 7% surveys. Thus when evaluating systems for the three emerging technologies studied, there is a clear tendency to do so in artificial rather than natural settings.

Of the 101 papers reporting from applied research with the purpose of engineering, only 44% *also* report evaluating the produced solutions. The majority of these, 66%, evaluations are done through laboratory experiments, 20% through field studies and the remaining 14% by surveys or normative writings. Of the 16 papers reporting *re-engineering* of systems, 56% also report evaluations. Of these, 89% are based on laboratory experiments and 11% use field studies. Thus when building new systems, there is a tendency towards *not* evaluating them, conversely when re-building them, evaluation is more prevalent.

2.5 Discussion

In response to the first research question, the literature review reveals a prevalent focus of current HCI research for emerging technologies and indicates a number of limitations and opportunities for future research related to this.

The literature review shows that, for the three emerging technologies studied, the focus of human-computer interaction research is on inventing interaction design by building new systems on the basis of good ideas and evaluating them in controlled environments. Supporting this focus, some attention is given to rebuilding existing systems and using field data to inform engineering and evaluation. The literature review shows a considerable lack of focus on natural setting research involving real use of technology in real contexts. Also, focus on understanding human-computer interaction for the technologies studied and construction of theory is almost absent.

One reason for this focus may be that applied research and laboratory experiments are simply easier to conduct and manage than field studies, case studies and action research, or that focusing on technology is simply easier than focusing on people.

Another reason may be that research into human-computer interaction for emerging technologies just *requires* a high level of experimental design for exploring innovative ideas and new opportunities offered by technology. Furthermore, it can be argued that emerging technologies are often simply not yet mature enough to be introduced into the real world and studied in natural settings (from a usability, technological as well as an economical point of view).

Regardless of these possible explanations, however, the identified focus of research on building systems ad-hoc and evaluating them in laboratory settings is inherently associated with a number of limitations. Especially worth mentioning are the two issues discussed below.

Firstly, given the prevalent applied approach to engineering, it seems implicitly assumed that we already know what systems to build and which specific problems to overcome. However, given the unclear properties, use, significance and implications of *emerging* technologies, this can hardly be true. On the contrary, it could be stated that human-computer interaction design for emerging technologies *in particular* requires research into what these technologies could be useful for, and how users perceive them in real use situations. The current focus on applied research for engineering systems does not address these questions much and consequently makes it difficult to set aside the implicit assumption that we already know what to build. The widespread use of laboratory-based evaluation only contributes further to this limitation by favoring focus on technology and implicitly assuming that context is less important.

Secondly, building and evaluating new systems on the basis of applied research and laboratory experiments typically results in very concrete conclusions about specific interaction design solutions. While such knowledge can be very important for future design of similar systems, it is often at a low level of abstraction and can be difficult to generalize. Focusing research on building and evaluating new systems without providing abstract reflections on the outcome thus limit the generation of theoretical understanding of the technologies studied. Consequently, this focus of research may produce results with only limited general value for human-computer interaction design.

The literature review and the limitations discussed above indicate a number of opportunities for future research in human-computer interaction design for emerging technologies.

Firstly, natural setting research into what systems to build and which specific problems to overcome would be a relevant contribution. Such research could obviously be informed by field studies exploring use contexts and evaluating systems but could also benefit from case studies on emerging technologies already being used in real world situations. Using action research methods for such studies could provide an opportunity to move experimentation with new technology out of the laboratories and at the same time grounding new ideas in real world situations. Such research would, however, demand a lot of resources. For example, it would require organizations to take up technologies that might be expensive and uncertain to succeed and accept researcher intervention over a long period of time. While survey research could be suitable for collecting large amounts of data from actual end users of emerging technologies and thus enabling wider generalizations of results, such research could be impeded by the limited propagation of some emerging technologies.

Secondly, research providing abstract and generalizable understanding of human-computer interaction for emerging technologies and descriptions of desirable properties of future systems would contribute greatly to the generation of cumulative theoretical knowledge. Apart from using more field studies for evaluations, such research could clearly benefit from case studies allowing a close study of technologies in real world contexts and enabling key issues to be identified, understood and described. Using action research for this would present an interesting opportunity to develop, evaluate and refine theoretical understanding of interaction design through studies of the use of emerging technologies in practice. The use of basic research could facilitate the development of theoretical frameworks to promote description and understanding. Also, basic research could examine the applicability of theories and methods from other disciplines.

3 Research Contributions

This chapter presents the five individual research contributions of the thesis (included in appendix A). The five contributions explore interaction design for virtual reality (immersion), mobile computer systems (mobility) and augmented reality (immersion *and* mobility) from the perspectives of *systems* and *processes* respectively.

The relation between the individual research contributions is outlined in table 3 below.

Emerging Technologies

Interaction Design

	Immersion	**Mobility**
Systems Concepts and interaction styles	1) Combinations of input devices and interaction styles for virtual reality display types	2) Combinations of input devices and interaction styles for handheld computer applications
	3) Development process and combinations of input devices and interaction styles for an augmented reality system	
Processes Methods and tools	4) Development process of a virtual reality application for the CAVE immersive display system	5) Development process of a handheld collaborative information system for a mobile use context

Table 3. Research contributions

1) Kjeldskov J. (2001) *Combining Interaction Techniques and Display Types for Virtual Reality.* In Proceedings of OzCHI 2001, Perth, Australia. Churchlands, Edith Cowan University Press.

2) Kjeldskov J. and Kolbe N. (2003) *Combining Input Devices and Interaction Styles for Handheld Computers.* Extended version of Kjeldskov J. and Kolbe N. (2002) *Interaction Design for Handheld Computers.* In Proceedings of the 5th Asian Pacific Conference on Human-Computer Interaction, APCHI 2002, Beijing, China. Beijing, Science Press.

3) Kjeldskov J. (2003) *Lessons from Being There: Interface Design for Mobile Augmented Reality.* In Andersen P. B. and Qvortrup L. (eds.) Virtual Applications: Applications With Virtual Inhabited 3D Worlds. Berlin, Springer-Verlag.

4) Kjeldskov J. and Stage J. (2003a) *Interaction Styles in Development Tools for Virtual Reality Applications.* In Halskov K. (ed.) Production Methods: Behind the Scenes of Virtual Inhabited 3D Worlds. Berlin, Springer-Verlag.

5) Kjeldskov J. and Stage J. (2003b) *The Process of Developing a Mobile Device for Communication in a Safety-Critical Domain.* In Proceedings of the 9th IFIP TC13 Conference on Human Computer Interaction, Interact 2003, Zürich, Switzerland. Zürich, IOS Press.

The sequence of the five paper contributions in this summary does not reflect the sequence of the conducted research.

3.1 Contribution 1

Combinations of input devices and interaction styles for virtual reality display types

Kjeldskov J. (2001) *Combining Interaction Techniques and Display Types for Virtual Reality.* In Proceedings of OzCHI 2001, Perth, Australia. Churchlands, Edith Cowan University Press.

The first paper explores concepts and interaction styles for an immersive emerging technology. The results reported originate from a qualitative usability study of interaction in virtual reality conducted over a period of six months. In this study, two different state-of-the art display systems: a six-sided CAVE measuring 2 ½ meters on each side and an 8x3 meter curved panoramic screen were used for displaying different virtual reality applications. For interaction with these applications, more than 40 combinations of input devices and interaction styles were implemented and qualitatively evaluated. Based on this, the paper presents a 3x12x2 matrix of interaction styles, input devices and display types populated by lessons learned from combinations of these.

The paper concludes that different types of virtual reality displays require different interaction design. Supporting this conclusion, the paper promotes two classes of concepts and outlines requirements for interaction styles suitable for human-computer interaction design for virtual reality.

The first class of concepts relates to immersion. Full and partial immersion in virtual reality are different user experiences supporting the feeling of "being in" and "looking at" a virtual environment respectively. Describing a display's potential to create an experience of immersion, the paper proposes the concept of *available field of view*. On the basis of this concept, virtual reality displays are divided into categories of *full immersive* displays (represented by e.g. six-sided CAVEs, head-mounted displays and boom-mounted displays) and *partial immersive* displays (represented by e.g. monitors/holobenches, large panoramic screens and 3-5 sided CAVEs).

The second class of concepts relates to interaction. Interaction design for virtual reality must support the user being present in a virtual environment. On the basis of this, the paper proposes that the concept of interaction is divided into three more specific categories of *orientating, moving* and *acting.* Orientating addresses the need for looking around in a virtual environment, moving addresses the need to move about in a virtual environment and acting addresses the need for selecting, moving, rotating and transforming virtual objects.

Interaction styles suitable for supporting orientating, moving and acting in virtual reality depend on the display's available field of view. Thus partial immersive displays require interaction styles supporting a relatively stationary user while interaction styles for full immersive displays must consider a more freely moving user. Exemplifying this, partial immersive displays require explicit support for orientating in a virtual environment, which can be realized for example by rotating the virtual world using a non-tracked 3D input device. In full immersive displays, this is an inherent property supported by head tracking.

3.2 Contribution 2

Combinations of input devices and interaction styles for handheld computer applications

Kjeldskov J. and Kolbe N. (2003) *Combining Input Devices and Interaction Styles for Handheld Computers.* Extended version of Kjeldskov J. and Kolbe N. (2002) *Interaction Design for Handheld Computers.* In Proceedings of the 5th Asian Pacific Conference on Human-Computer Interaction, APCHI 2002, Beijing, China. Beijing, Science Press.

The second paper explores concepts and interaction styles for a mobile emerging technology by studying which interaction styles fit which input devices on handheld computers and why. The results reported are based on a quantitative usability evaluation of three experimental prototypes combining traditional interaction styles described in the HCI literature with input devices commonly available on handheld computers. Based on this, the paper presents two 3x3 matrices of input devices and interaction styles describing effectiveness and homogeneity of user performance and a series of guidelines for mobile computer system interaction design.

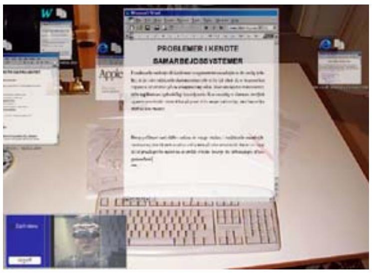
The paper concludes that different combinations of input devices and interaction styles for handheld computers influence the effectiveness and consistency of user performance. Supporting this conclusion, the paper promotes two classes of concepts and outlines requirements for interaction styles suitable for human-computer interaction design for mobile computer systems.

The first class of concepts relates to mobility. To support mobility, screen size on mobile computer systems is reduced. On handheld computers, it is found that this reduction results in cursor keys and direct pointers being equally suited for menu selection and direct manipulation. Explaining this finding, the concept of *limited interaction space* is proposed for describing the reduced interface complexity of mobile computer systems.

The second class of concepts relates to interaction. Handheld computers have limited input and output capabilities. Unfolding this problem, the paper proposes the concepts of *level of versatility* for classifying input devices in relation to their restriction on interaction and *level of required input* for classifying interaction styles in relation to the input necessary to operate them. On the basis of this, function keys, cursor keys and direct pointers are classified as having low, medium and high level of versatility respectively. Similarly, form filling is classified as requiring high level of input while menu selection and direct manipulation require only low.

Interaction styles suitable for mobile computer systems depend on their level of required input in combination with the level of versatility of potential input devices within the limitations of the interaction space defined by the display size. Thus, for example, the value of pen-based direct pointers for menu selection and direct manipulation on mobile computer systems can be questioned, and the use of cursor keys be suggested as a possible alternative.

3.3 Contribution 3



Development process and combinations of input devices and interaction styles for an augmented reality system

Kjeldskov J. (2003) *Lessons from Being There: Interface Design for Mobile Augmented Reality.* In Andersen P. B. and Qvortrup L. (eds.) Virtual Applications: Applications With Virtual Inhabited 3D Worlds. The Staging Book Series Vol. 3. Berlin, Springer-Verlag.

The third paper explores concepts, interaction styles, methods and tools for a mobile immersive emerging technology. The results reported are based on lessons learned from the process of developing an augmented reality system. In this process, a field study provided rich data about interaction in physical space and a design patterns method informed interaction design. The system was implemented in a multimedia development tool. Based on this, the paper describes the development process, presents six design patterns and illustrates the design of the system.

The paper concludes that interaction design for augmented reality can be based on principles of interaction in physical space. Supporting this conclusion, the paper promotes one central concept and outlines requirements for suitable interaction styles. Furthermore, the paper concludes that the process of designing human-computer interaction for augmented reality is supported by methods providing knowledge about interaction in physical space and tools providing advanced programming environments.

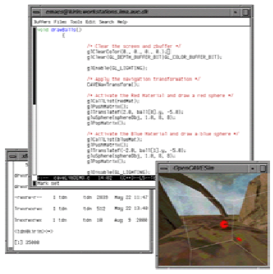
The central concept proposed relates to immersion and mobility. Augmented reality facilitates the user being immersed in virtual space and mobile in physical space. Signifying the relation between the two, the paper promotes the concept of *level of contextual detail* describing the responsiveness of virtual space to contextual changes in physical space.

Interaction styles suitable for augmented reality relate interaction in physical and virtual space by applying a high level of contextual detail. Exemplifying this, the presented system makes use of motion tracking and datagloves for interaction styles *relating physical and virtual space,* and supporting *mutual focus in communication*.

Designing interaction for augmented reality requires methods providing knowledge about interaction in physical space. Meeting this challenge, the paper shows that field studies are useful for providing rich and detailed data about the use context. Providing an analytical abstraction of the field data, the design pattern method proved useful for capturing interaction in physical space and informing the form of creative design solutions.

Finally, tools support the process of designing human-computer interaction by shortening the distance between development and target platforms and providing advanced programming environments. Exemplifying this, the multimedia tool employed facilitated fast iterations with easy and accurate previews because it ran on the target platform. Furthermore the tool facilitated creative design of interaction styles through low-level access to data from input devices and advanced programming facilities for exploiting those data.

3.4 Contribution 4



Development process of a virtual reality application for the CAVE immersive display system

Kjeldskov J. and Stage J. (2003a) *Interaction Styles in Development Tools for Virtual Reality Applications.* In Halskov K. (ed.) Production Methods: Behind the Scenes of Virtual Inhabited 3D Worlds. The Staging Book Series Vol. 4. Berlin, Springer-Verlag.

The fourth paper explores tools supporting the design of interaction for an immersive emerging technology. The results reported originate from an experimental development process in which two independent developers implemented two identical virtual reality systems for a six-sided CAVE using a command language and a direct manipulation based tool respectively. During the experiment, the developers documented their work in diaries and recorded the time spent on reaching a number of specific milestones. Based on this, the paper describes how the applied tools supported interaction design for virtual reality.

The paper concludes that the process of designing human-computer interaction for virtual reality is supported by tools providing accurate previews of the system and advanced programming environments. These conclusions are supported by two central findings described below.

Firstly, systems for the CAVE are not developed in the CAVE itself but on desktop workstations. Hence designing interaction for virtual reality involves a distance between development and target platforms. This consequently challenges facilities for previewing the system during development. In the tool employing command language, this was supported by running the system in a CAVE simulator and worked fine for minor debugging purposes. However, running the system in the real CAVE often required modifications due to inaccuracy of the simulator preview and limited portability of code between different hardware platforms. This limited the frequency of iterations. In the tool employing direct manipulation, previewing was seamlessly integrated into the design window and accurately represented the performance of the system in the CAVE. This facilitated fast implementation of a preliminary prototype, required fewer adjustments to the final system and increased the frequency of iterations.

Secondly, designing interaction for virtual reality involves the creation of a three-dimensional virtual environment in which the user can orientate, move and act. This requires complex programming. In the command language tool, this was supported by an advanced programming environment (C and C++), which facilitated huge freedom over the functionality of the system and design of interaction styles. In the direct manipulation tool, the programming environment was limited to predefined events and actions without facilities for “real” programming. This was very time consuming to use and severely hindered the design process during programming intensive phases. Also, the lack of programming facilities constrained the design of interaction styles.

3.5 Contribution 5

Development process of a handheld collaborative information system for a mobile use context

Kjeldskov J. and Stage J. (2003b) *The Process of Developing a Mobile Device for Communication in a Safety-Critical Domain.* In Proceedings of the 9th IFIP TC13 Conference on Human Computer Interaction, Interact 2003, Zürich, Switzerland. Zürich, IOS Press.

The fifth and final paper explores methods and tools supporting the design of interaction for a mobile emerging technology. The results reported originate from the process of developing a handheld collaborative information system for a safety-critical domain. In this process, a field study provided rich data about a mobile work activity and an object-oriented method informed interaction design. The system was implemented in a development tool for mobile computer systems and evaluated in a high-fidelity simulation of the real use context. Based on this, the paper describes the development process, presents the outcome of the analysis, illustrates the system, and reports on the evaluation.

The paper concludes that the process of designing human-computer interaction for mobile computer systems is supported by methods providing knowledge about mobile work activities and tools facilitating easy portability of code to the target platform during development. These conclusions are supported by two central findings described below.

Firstly, mobile computer systems facilitate new situations of computer technology use. Envisioning future use of mobile computer systems, however, requires methods for acquiring detailed insight into potential areas of application. In the reported development process, field studies provided insight into an existing work activity and revealed a number of limitations in current practice. Some of these could potentially be overcome by mobile computer systems. Providing an analytical abstraction of the field data, the object-oriented method proved useful for capturing structures and relations in work activities and communication. The outcome of the object-oriented analysis effectively informed interaction design by specifying requirements for functionality and indicating viable form.

Secondly, systems for mobile computers are not developed on mobile computers themselves but on desktop workstations. Designing interaction for mobile computer systems therefore involves a distance between development and target platforms challenging the facilities for previewing systems during development. In the development tool employed, this was supported not only by providing accurate preview facilities on the development platform but also by easy portability of code to the target platform. This facilitated fast and frequent design iterations and resulted in few adjustments to the final system being required. Although it provided an advanced programming environment (embedded Visual Basic), limitations of this tool, constrained interaction design to the use of conventional interaction styles such as form-filling and menu selection.

4 Research Methods

This chapter presents and elaborates on the research design applied for enquiring into the three research questions stated in Chapter 1. The conceptual foundation for this is the discussion of research methods and research purposes presented in Chapter 2.

Choosing a method of research is not as much a question of making a right choice as it is a question of making an appropriate one (Galliers and Land 1987). The challenge of choosing appropriate research methods consists of understanding the potentials and limitations of different approaches and choosing between these in light of a specific research focus and interest.

The research of this thesis involves four different research methods consisting of both natural, artificial and environment independent setting studies. The choice of research methods for addressing the three individual research questions of the thesis is outlined in table 4 below.

Research Question	Research Object	Research Method	Research Purpose	Research Setting
1) Focus of Research	Literature	Survey research	Describing Understanding	Environment independent
2) Concepts and interaction styles	Systems	Laboratory experiments	Evaluating Understanding	Artificial
3) Methods and tools	Processes	Laboratory experiments Field studies	Engineering Understanding	Artificial Natural

Table 4. Methods applied in response to the three research questions

The strengths and limitations of this research design are discussed below.

4.1 Literature: Focus of Research

According to Galliers (1992), reviewing past research should ensure that future research builds on past endeavors. In response to this, research question #1 about the focus of HCI research for emerging technologies is addressed through a literature survey with the purpose of describing and understanding current practice. By classifying research papers in a predefined matrix of methods and purpose, overall tendencies in current research are uncovered. The research papers were selected from a broad range of conference proceedings and journals and classified by two researchers independently.

One of the disadvantages of using a survey method for addressing the question of HCI research focus is related to the coding and analysis of data. While the classification of research papers in a predefined matrix is not done to test hypotheses, the matrix still constrains the coding of data and thus limits the space for conclusions emerging from the study. Also, the review is subject to the assumption that the categories of the matrix are suitable for classifying HCI research. This is not necessarily the case. Using a grounded research approach might alternatively overcome the limitation of the predefined matrix and facilitate a more inductive and theoretical account of present research focus to be developed (Myers 1997). However, it would also be more resource demanding.

Another disadvantage is that surveying literature only provides limited insight on the causes behind the identified focus of research (Galliers 1992). Also, the survey is conditional on the assumption that publications are representative for the focus of a research field. Alternatively, basing the survey on questionnaires or interviews with key researchers could provide more direct insight into current practice.

The advantages of using a survey method are that it enables the collection of a large and generalizable data sample in an environment independent setting and at a relatively low cost in terms if time and effort. Also, it facilitates an overview, which would be difficult to obtain by other methods, for example field studies (Wynekoop and Conger 1990).

4.2 Systems: Concepts and Interaction Styles

Research question #2 about concepts and interaction styles is addressed through three studies applying different laboratory experiments with the purpose of understanding interaction design through evaluation of systems.

One of the disadvantages of the laboratory experiment method is that results from artificial setting research can be difficult to generalize to real world situations (Wynekoop and Conger 1990). This is a well-known limitation within HCI research (Shneiderman 1998:132, Dix et al. 1998:407) but is emphasized further by technologies such as virtual reality, augmented reality and mobile computer systems because they involve specialized contexts of use. Applying a natural setting research method like for example case or field studies may overcome this limitation and provide more realistic data (Brewster 2002, Nielsen 1998). However, natural setting HCI research is difficult and often suffers from limited control and complicated data collection (Sawhney and Schmandt 2000, Pascoe et al. 2000).

Other disadvantages of laboratory experiments originate from different applications of the method in practice (Wynekoop and Conger 1990). For the research contributions of this thesis, the laboratory experiments employed different levels of experimental control, different manipulation of variables, and different types of data collection.

In the first study (paper contribution 3), the laboratory experiment employed a narrow focus on interaction design for one specific system. The experiments were highly explorative, and controlled manipulation of variables was low. The collected data was qualitative and the sample size was small. The outcome of such studies has limitations similar to those of a case study: conclusions are restricted to single events, and the use of them in other situations is impeded by the study's unique character (Galliers 1992).

In light of this, the second study (paper contribution 1) compared a larger number of interaction designs and enforced more experimental control through selection of test subjects and controlled manipulation of variables. Again, qualitative data were collected and the sample size was small. The outcome of this study is more generalizable and has broader scope.

Contrasting the first two studies, the third study (paper contribution 2) employed a laboratory experiment with quantitative data collection on the basis of a few manipulated variables and a larger sample size. The outcome of such studies is a statistically computable, comparable and generalizable dataset useful to draw objective conclusions from. However, a disadvantage of such studies is the risk of measuring something that is only weakly related to the phenomena of interest (Nielsen 1993).

The advantage of using a laboratory experiment method for studying interaction design is that it enables controlled manipulation of variables and precise measurements of phenomena of interest, cf. (Wynekoop and Conger 1990). Consistently applying the same research method across different technologies contributed to comparability between studies. Also, in the studies collecting qualitative data on the basis of a few cases, this provided rich insight into the studied interaction design phenomena, cf. (Strauss and Corbin 1998).

4.3 Processes: Methods and Tools

Research question #3 about methods and tools is addressed through three studies applying laboratory experiments for the purpose of understanding the process of interaction design by engineering prototype systems.

Laboratory experiments can take different forms depending on, among other factors, experimental design, participants and the role of the researcher (see e.g. Preece et al. 1994, Nielsen 1993). In the experiments with interaction design processes presented in this thesis, two different approaches were adopted: 1) a controlled experiment with researchers strictly being observers and 2) two experimental design processes with researchers being actively involved in system development.

In the first study (paper contribution 3), an explorative experiment investigated the use of methods and tools for supporting interaction design. The studied design process lasted six months and involved field studies, analysis of empirical data and design and implementation of a prototype system. This was all conducted by researchers while also studying the design process itself.

Contrasting this, the second study (paper contribution 4) compared the use of two different tools for implementing the same interaction design in a controlled laboratory experiment. The tools were assigned to two independent developers, who were given the task of implementing a specified system within two weeks. The collected data was quantitative in the form of diaries documenting, among other information, time spent on meeting 14 predefined milestones. The researchers observed the design processes but did not intervene or participate.

In the third study (paper contribution 5), the experiment was again explorative. The studied design process lasted twelve months during which time researchers conducted field studies, analyzed empirical data and implemented a prototype system while also studying their use of methods and tools for doing so.

The advantage of using a laboratory experiment method for studying design processes is that it facilitates controlled manipulation of variables and focuses on specific phenomena of interest, cf. (Wynekoop and Conger 1990). The use of field studies contributed to the laboratory experiments by providing real-life context, cf. (Galliers 1992) and by informing interaction design. While in the first and third study, analyzing our own use of methods and tools did not provide insight into the use of the same methods and tools in the hands of real developers, the explorative experimental approach benefited from advantages similar to those of action research. The researchers could intervene with the process and gained first-hand insight into the situation (Wynekoop and Conger 1990). Although the design processes compared in the second study were simplified in scope and the developed systems were detached from real use contexts, the controlled experimental approach facilitated objective comparisons between the performances of different tools throughout the process.

4.4 Summary

In summary, the research design of the thesis has involved survey research, laboratory experiments and field studies. The primary weaknesses of this design are the classification process of the literature review and the limited generality to real world situations of the experiments with systems and processes. The primary advantages are the large sample size of the survey and the experimental control and precise focus of the experiments.

5 Conclusions

This thesis deals with human-computer interaction design for three emerging technologies: virtual reality, augmented reality and mobile computer systems. It is based on experimental design and evaluation of five implemented prototype systems. In addition to this, it includes a literature survey of 220 research papers. This chapter summarizes the conclusions of the presented research structured in accordance with the three research questions stated in Chapter 1. Moreover, it points out limitations and indicates avenues for further research.

5.1 Literature: Focus of Research

The first research question addresses the *focus of research* in HCI for the three studied emerging technologies. The literature review reveals that:

1) For the three emerging technologies studied, the focus of human-computer interaction research is on inventing interaction design by building new systems ad-hoc and evaluating them in controlled environments. Some attention is also given to rebuilding existing systems and using field data to inform engineering and evaluation. Real use contexts are rarely studied.

2) Focusing on building new systems ad-hoc and evaluating them in artificial settings limits the generation of theoretic knowledge and makes it difficult to set aside the assumption that we already know what systems to build and which specific problems to overcome.

3) The immediate opportunity for future research is to use more natural setting research for both building and evaluating systems and to focus on generating abstract and generalizable knowledge about human-computer interaction design for the emerging technologies studied.

In conclusion, the current focus of research for the three emerging technologies studied is on building and evaluating systems ad-hoc. This limits the generation of cumulative theoretical knowledge. Opportunities for further research include more natural setting research and increased focus on producing theoretical contributions.

5.2 Systems: Concepts and Interaction Styles

The second research question addresses *concepts* and *interaction styles*. Studying the interaction designs of the implemented systems reported in papers 1, 2 and 3 reveal that:

1) For the three emerging technologies studied, concepts suitable for human-computer interaction design fall into two classes. First, the concepts of *available field of view, limited interaction space* and *level of contextual detail* describe unique properties of the studied technologies related to immersion and mobility. Secondly, the concepts of *orientating, moving and acting*, *level of versatility* and *level of required input* describe challenges of the technologies related to interaction. These concepts emphasize and describe issues not addressed previously by concepts for human-computer interaction

with traditional desktop computers. This indicates that interaction design for emerging technologies requires development of new concepts to supplement existing ones.

2) For the three emerging technologies studied, suitable interaction styles respond to the challenges of each technology by exploiting its unique properties. For virtual reality, suitable interaction styles support orientating, moving and acting by exploiting the display's available field of view. For mobile computer systems, suitable interaction styles balance the level of versatility of input devices and the level required input of interaction styles by utilizing the limited interaction space. For augmented reality, suitable interaction styles relate interaction in physical and virtual space by applying a high level of contextual detail. For virtual and augmented reality, this results in new interaction styles such as motion tracking, and the refinements of existing ones such as dataglove-based 3D direct manipulation. For mobile computer systems, traditional interaction styles such as form filling and menu selection are still useful but must be re-considered in the light of available input devices and limited screen size.

In conclusion, interaction design for the three emerging technologies studied requires the development of new concepts describing their unique properties and challenges, and the development of new interaction styles or refinement of existing ones.

5.3 Processes: Methods and Tools

The third research question addresses *methods* and *tools*. Studying the processes of designing interaction for the implemented systems reported in papers 3, 4 and 5 reveals that:

1) For the three emerging technologies studied, designing interaction involves new use situations being considered and investigated and requires methods for supporting this. Elements of methods exist, which are useful for supporting different aspects of designing interaction for these technologies but they are fragmented and do not support the process as a whole. For both augmented reality and mobile computer systems, field studies supported the design process by providing rich data about use contexts. For augmented reality, analyzing the field data using a design pattern method captured patterns of interaction in physical space. For mobile computer systems, analyzing the field data using an object-oriented method captured structures and relations in work activities and communication. While both methods informed creative design, the design patterns method provided more support for the designer on what the system should look like (form) than on what it should do (functionality). Conversely, the object-oriented method provided more support on functionality than on form.

2) None of the tools studied supported the design process as a whole by facilitating seamless integration between analysis, design and implementation. Nor did they provide explicit facilities for responding to unique properties and challenges of a technology. For example, the tool for developing mobile computer systems did not facilitate distribution of the system among physically dispersed users and the virtual reality development tools did not include facilities for specifying the properties of the display type and input devices to be used. However, tools with good preview facilities and easy portability of code to the target platform supported fast and frequent iterations. Tools with advanced programming environments supported complex implementation and development of creative interaction design.

In conclusion, for the three emerging technologies studied, methods are fragmented and do not support the process of designing interaction as a whole. Elements of different methods are useful when they provide analytical abstractions of the use context informing form and functionality. None of the tools employed supports the process as a whole either but are useful for interaction design when they bridge the gulf between development and target platforms and provide complex programming environments.

The conclusions above show that human-computer interaction design for virtual reality, augmented reality and mobile computer systems are fields of research characterized by 1) experimentation with new technology rather than studies of real world use, 2) systems requiring the development of new concepts and interaction styles or refinements of existing ones and 3) design processes requiring the refinements of existing methods and tools or development of new ones. While interaction styles suitable for interaction design are closely related to suitable concepts, there is presently no relation between the aspects of the design process supported well by methods and those supported well by current tools.

5.4 Emerging Technologies

The research presented in this thesis shows that the four themes of concepts, interaction styles, methods and tools adopted from traditional HCI research are also relevant for the study of human-computer interaction design for emerging technologies. However, the themes result in different content and requirements across technologies because these involve different specific challenges. Virtual reality challenges existing knowledge about human-computer interaction because it involves a fundamental shift from traditional conceptions of computer applications as software running on machines populating our desktops, toward a space the user can physically enter and be completely surrounded by. Mobile computer systems challenge existing knowledge about human-computer interaction because mobility in physical space facilitates new situations of computer use in which people may be primarily engaged in activities in the surrounding environment. Augmented reality challenges human-computer interaction design in the same ways as virtual reality and mobile computer systems because it involves the user being simultaneously immersed in virtual space and mobile in physical space, but also involves unique challenges because of the very close relation between the two.

Virtual reality, augmented reality and mobile computer systems constitute different directions of emergence within computing. In the introduction, the three studied technologies were plotted on a linear continuum from the real to the virtual environment (Milgram and Colquhoun 1999) with mobile computer systems and virtual reality populating the extremes and augmented reality located in between. Based on the research presented in this thesis, however, I argue that this representation is too simplistic. The problem with the linear continuum is that it presents immersion into virtual space and mobility in physical space as mutually exclusive and leaves no space for traditional desktop computers. Instead I propose that immersion and mobility are independent, orthogonal dimensions. In figure 4 below I use these dimensions to create a technology space that can be populated by the three emerging technologies studied in this thesis with reference to traditional desktop computers.

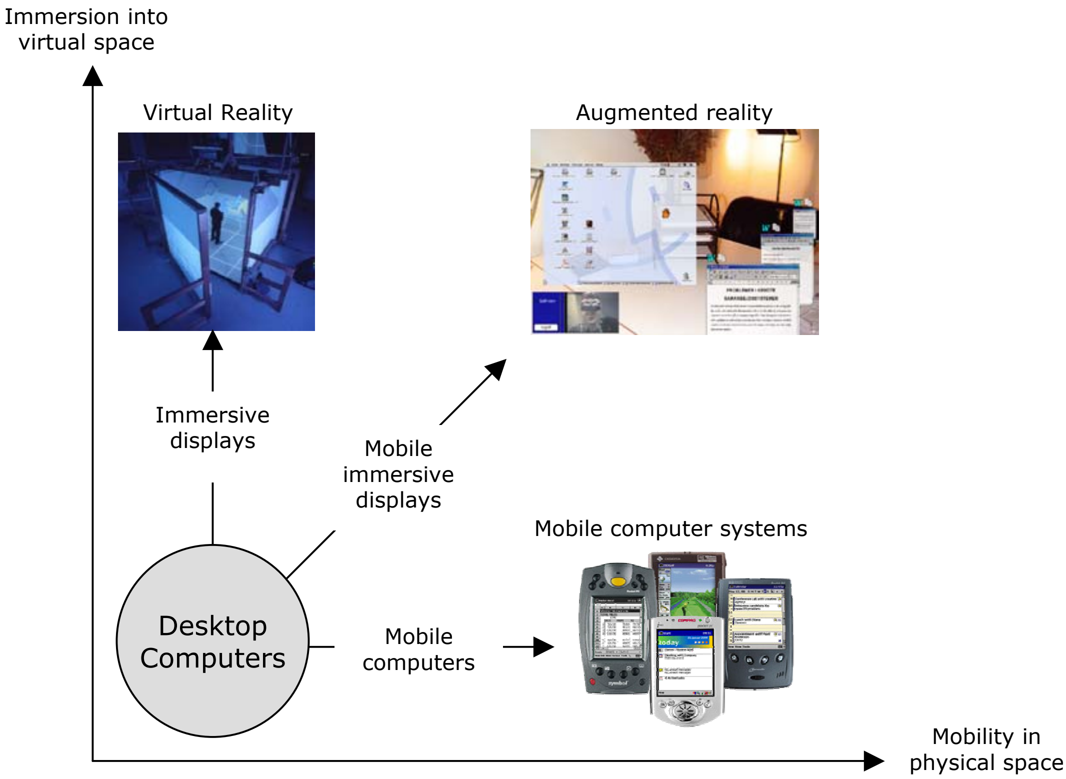


Figure 4. Immersion and mobility as orthogonal dimensions of emerging technologies

The population of the technology space depicted in figure 4 signifies relations between different emerging technologies. Thus, for example, technologies aligned horizontally or vertically within a given dimension of emergence, like for example virtual reality and augmented reality, share properties and challenges rising from this direction of development. When two emerging technologies share properties and challenges, it is also likely that interaction design for these technologies can benefit from knowledge about concepts, interaction styles, methods and tools for each other. Conversely, diagonal technologies, like for example virtual reality and mobile computer systems, are very different. Consequently such technologies will most likely benefit little from HCI knowledge about each other.

As immersion and mobility constitute only two of many possible directions of emergence, as discussed in the introduction, figure 4 could be extended with other dimensions such as multi-modality, context-awareness and embeddedness, allowing other technologies to be plotted into the technology space.

5.5 Limitations

The research of this thesis has a number of limitations. Firstly, the research approach has been to investigate human-computer interaction design across a number of emerging technologies. Due to the constraints of the thesis, this means that only a few studies have been carried out in relation to each technology. While broadening the research, this has the consequence that the research into each of the individual emerging technologies has reduced depth in comparison to repeated studies of only one technology. Another limitation of the research approach is the divided focus on both systems and processes. Again, this has increased the breadth of the research but at the cost of depth into one aspect of interaction design.

Secondly, the thesis lacks real-world evaluation of its findings. The proposed concepts and interaction styles and suggestions for methods and tools originate from laboratory experiments and have not been evaluated in natural settings. This limits the general validity of the conclusions. Also, the study on design processes for virtual reality focused only on tools and did not involve studies of methods.

Finally, the categories of methods and purposes in the research review can be criticized for being vague and overlapping. Also, it is debatable whether the eight categories of methods belong to the same level of abstraction or if some categories could be subordinated others. Combined with the fact that many research papers provide only limited information about method, this means that it can be difficult to decide which category a specific paper belongs to. The classification of literature thus relies on the researchers comprehension of the categories and ability to make a qualified judgment on the basis of sometimes scarce information.

5.6 Future Research

The research focus, findings and limitations of this thesis leave many opportunities for future research.

Firstly, this thesis has addressed human-computer interaction *design* for emerging technologies. However, as stated in the introduction, this only constitutes one of several areas of HCI research. Other areas include implementation, evaluation and user studies. While implementation of human-computer interaction design for emerging technologies is already well covered in existing research (as pointed out in the literature review), future research should address the development of suitable evaluation techniques for emerging technologies and study phenomena surrounding the use of them.

Secondly, in response to the limitations discussed above, future research should subject the three emerging technologies studied to more thorough research of human-computer interaction design. This research should strive to evaluate the validity of the presented findings through similar studies and seek to enhance and refine these through complementary approaches and alternative focal points. It could, for example, be relevant to address more thoroughly the concepts of immersion and mobility and their influence on human-computer interaction design.

Thirdly, future research should involve other emerging technologies, identifying their own unique properties and challenges for human-computer interaction design and assessing the generality of the presented findings for the three technologies studied in relation to the overall class of *emerging technologies.*

Finally, future research should attempt to include studies of real world systems and development processes. This would provide insight on the use of the studied technologies by real users in relation to real contexts and thereby complement the present prevalent focus on experimental systems in artificial settings.

References

Andersen H. and Kaspersen L. B. (1996) *Klassisk og Moderne Samfundsteori* (Classic and Contemporary Social Theory) (in Danish). Copenhagen, Hans Reitzels Forlag

ACM SIGCHI (1992) *Curricula for Human-Computer Interaction.* ACM Special Interest Group on Computer-Human Interaction Curriculum Development Group. New York, ACM

Azuma R. (1997) A Survey of Augmented Reality. In Presence: Teleoperators and Virtual Environments. Vol. 6(4), 355-385

Basili V. R., Selby R. W. and Hutchins D. H. (1986) Experimentation in software engineering. IEEE Transactions on Software Engineering, SE-12, 733-743

Behringer R., Klinker G., Mizell D. W. (eds.) (1999) *Augmented Reality: Placing Artificial Objects in Real Scenes.* Natick, A K Peters

Benbasat I. (1985) An analysis of research methodologies. In MacFarlan F.W. (Ed.) *The Information System Research Challenge.* Boston, Harvard Business School Press, 47-85

Bergman E. (ed.) (2000) *Information Appliances and Beyond: Interaction Design for Consumer Products.* San Francisco, Morgan Kaufmann Publishers

Beyer H. and Holtzblatt K. (1998) *Contextual Design: Defining Customer-Centered Systems.* San Francisco, Morgan Kaufmann Publishers

Billinghurst M., Kato H. and Poupyrev I. (2001) The MagicBook: a transitional AR interface. Computers and Graphics (2001) Vol. 25, 745-753

BMW (2002) BMW Assist. http://www.bmw.com/bmwe/products/equip/telematics/

Bohnenberger T., Jameson A., Kruger A. and Butz A. (2002) Location-Aware Shopping Assistance: Evaluation of a Decision-Theoretic Approach. In Proceedings of Mobile HCI'02, Pisa, Italy. Berlin, Springer-Verlag

Borchers J. (2001) *A Pattern Approach to Interaction Design.* New York, John Wiley & Sons Ltd

Brewster S. (2002) Overcoming the Lack of Screen Space on Mobile Computers. Personal and Ubiquitous Computing (2002), Vol. 6, 188-205

Bryson S. (1996) Virtual Reality in Scientific Visualization. Communications of the ACM, Vol. 39(5), 62-71

Burton (2003) Burton Amp Jacket. http://www.burton.com/gear/pr_amp_jacket.asp

Carroll J. M. (ed.) (2002) *Human-Computer Interaction in the New Millennium.* Boston, Addison Wesley and New York, ACM Press

Chlamtac I. and Redi J. (1998) Mobile Computing: Challenges and Potential. In *Encyclopedia of Computer Science.* 4th Edition. International Thomson Publishing

Conway M., Audia S., Burnette T., Cosgrove D., Christiansen K., Deline R., Durbin J., Gossweiler R., Koga S., Long, C., Mallory B., Miale S., Monkaitis K., Patten J., Pierce J., Shochet J., Staack D., Stearns B., Stoakley R., Sturgill C., Viega J., White J., Williams G. and Pausch R. (2000) Alice: Lessons Learned from Building a 3D System For Novices. In Proceedings of CHI'00, The Hague, Netherlands, ACM

Cunningham S. (1998) Outside the Box: The Changing Shape of the Computing World. SIGCSE Bulletin Vol. 30(4), 4-7

Dahlberg P. (2003) *Local Mobility.* Ph.D. Thesis, Department of Informatics, Göteborg University. Gothenburg Studies in Informatics Report 27, ISSN 1400-741X

Dey A. K. and Abowd G. D. (2000) Towards a Better Understanding of Context and Context-Awareness. In Proceedings of the 2000 Conference on Human Factors in Computing Systems, CHI'00. Workshop on The What, Who, Where, When, and How of Context-Awareness. The Hague, Netherlands, ACM

Dix A., Finlay J. E., Abowd G. D. and Beale R. (1998) *Human-Computer Interaction.* 2nd Edition. London, Prentice Hall Europe

Electrolux (1999) Electrolux Screen Fridge. http://www.electrolux.com/screenfridge/

Feiner S., MacIntyre B. and Höllerer T (1999) Wearing It Out: First Steps Toward Mobile Augmented Reality Systems. In Ohata Y. and Tamura H. (1999) *Mixed Reality: Merging Real and Virtual Worlds.* Tokyo, Ohmsha Ltd

Galliers R. D. (1992) Choosing Information Systems Research Approaches. In Galliers R. D. (Ed.) (1992) *Information Systems Research: Issues, Methods and Practical Guidelines.* Boston, MA, Blackwells Scientific Publications, 144-162

Galliers R. D. and Land F. F. (1987) Choosing Appropriate Information Systems Research Methodologies. Communications of the ACM, Vol. 30(11), 900-902

Gellersen H., Beigl M. and Schmidt A. (1999) Environment-Mediated Mobile Computing. In Proceedings of the 1999 ACM Symposium on Applied Computing, SAC'99, ACM

Greenbaum J. and Kyng K. (eds.) (1992) *Design at work: Cooperative Design of Computer Systems.* London, Lawrence Erlbaum Associates, Inc., Publishers

Johnson W. W., Battiste V., and Holland S. (1999) A cockpit display designed to enable limited flight deck separation responsibility. In Proceeding of the 1999 World Aviation Conference, Anaheim, CA

Kjeldskov J. and Kolbe N. (2003) Combining Input Devices and Interaction Styles for Handheld Computers. Extended version of Kjeldskov J. and Kolbe N. (2002) Interaction Design for Handheld Computers. In Proceedings of the 5th Asian Pacific Conference on Human-Computer Interaction, APCHI 2002, Beijing, China. Beijing, Science Press

Kjeldskov J. and Stage J. (2003b) The Process of Developing a Mobile Device for Communication in a Safety-Critical Domain. In Proceedings of the 9th IFIP TC13 Conference on Human Computer Interaction, Interact 2003, Zürich, Switzerland. Zürich, IOS Press

Kjeldskov J. and Stage J. (2003a) Interaction Styles in Development Tools for Virtual Reality Applications. In Halskov K. (ed.) *Production Methods: Behind the Scenes of Virtual Inhabited 3D Worlds*. Berlin, Springer-Verlag

Kjeldskov J. (2003) Lessons from Being There: Interface Design for Mobile Augmented Reality. In Andersen P. B. and Qvortrup L. (eds.) *Virtual Applications: Applications With Virtual Inhabited 3D Worlds.* Berlin, Springer-Verlag

Kjeldskov J. (2001) Combining Interaction Techniques and Display Types for Virtual Reality. In Proceedings of OzCHI 2001, Perth, Australia. Churchlands, Edith Cowan University Press

Kristoffersen S. and Ljungberg F. (1999) Mobile Use of IT. In Proceedings of the 19th Information Systems Research Seminar in Scandinavia. Finland

Laurel B. (1992) *Computers as Theatre.* New York, Addison-Wesley

Lewis I. M. (1985) *Social Anthropology in Perspective.* Cambridge, Cambridge University Press

Mathiassen L. M., Munk-Madsen A., Nielsen P. A. and Stage J. (2001) *Object-Oriented Analysis and Design.* Aalborg, Marko

Milgram P. and Colquhoun Jr. H. (1999) A Taxonomy of Real and Virtual World Display Integration. In Ohata Y. and Tamura H. (eds.) (1999) *Mixed Reality: Merging Real and Virtual Worlds.* Tokyo, Ohmsha Ltd

MIT (2003) Emerging Technologies Conference. MIT Magazine of Innovation. http://www.etc2003.com/

Mobile HCI (2002) The 1st International Conference on Human Computer Interaction with Mobile Devices. http://giove.cnuce.cnr.it/mobilehci02.html

Moran T. P. (1981) The Command Language Grammar: a Representation for the User Interface of Interactive Systems. International Journal of Man-Machine Studies, Vol. 15(1), 3-50

Murray J. (1997) *Hamlet on the Holodeck: The Future of Narrative in Cyberspace*. Cambridge, MA, The MIT Press

Myers M. D. (1997) Qualitative Research in Information Systems. MIS Quarterly, Vol. 21(2), 241-242

Newman W. M. and Lamming M. G. (1995) *Interactive System Design.* Workingham, Addison-Wesley

Nielsen C. (1998) Testing in the Field. In Proceedings of the 3rd Asia Pacific Computer Human Interaction Conference. Werner B. (ed.), IEEE Computer Society, California, 285-290

Nielsen J. (1993) *Usability Engineering.* San Francisco, Morgan Kaufmann Publishers

Norman D. A. and Draper S. W. (eds.) (1986) *User Centered System Design.* London, Lawrence Erlbaum Associates

Norman D. A. (1988) *The Design of Everyday Things.* New York, Doubleday

Norman D. A. (1998) *The Invisible Computer: why good products can fail, the personal so complex, and information appliances are the solution.* Cambridge, MA. The MIT Press

Ohata Y. and Tamura H. (eds.) (1999) *Mixed Reality: Merging Real and Virtual Worlds.* Tokyo, Ohmsha Ltd

Oviatt S. Cohen P., Wu L., Vergo J., Duncan L., Suhm B., Bers J., Holzman T., Winograd T., Landay J., Larson J. and Ferro D. (2002) Designing the User Interface for Multimodal Speech and Pen-Based Gesture Applications: State-of-the-Art Systems and Future Research Directions. In Carroll J. M. (ed.) (2002) *Human-Computer Interaction in the New Millennium.* Boston, Addison Wesley and New York, ACM Press

Oxford Advanced Learner's Dictionary (1995) Fifth Edition. Edited by Jonathan Crowther. Oxford, Oxford University Press

Pascoe J., Ryan N. and Morse D. (2000) Using While Moving: HCI Issues in Fieldwork Environments. ACM Transactions on Computer-Human Interaction, Vol. 7(3), 417-437

Perry M., O'Hara K., Sellen A., Brown B. and Harper R. (2001) Dealing with Mobility: Understanding Access Anytime, Anywhere. ACM Transactions on Computer-Human Interaction, Vol. 8(4), 323-347

Preece J., Y. Rogers H., Sharp D., Benyon D., Holland S. and Carey T. (1994) *Human-Computer Interaction.* Workingham, Addison-Wesley

Rapoport R. N. (1970) Three Dilemmas in Action Research. Human Relations, Vol. 23(4), 499-513

Raskin J. (2000) The Humane Interface: New Directions for Defining Interactive Systems. Boston, Addison-Wesley

Roberts J. (2002) Is there a Next Big Thing?. The Information Age. http://www2.idg.com.au/infage1.nsf/all/F5A4077159B9DF5ACA256B9C0019517B?OpenDocument

Rheingold H. (1992) *Virtual Reality: The revolutionary Technology of Computer-Generated Artificial Worlds and How it Promises to Transform Society.* Touchstone Books

Sawhney N. and Schmandt C. (2000) Nomadic Radio: Speech and Audio Interaction for Contextual Messaging in Nomadic Environments. ACM Transactions on Computer-Human Interaction, Vol. 7(3), 353-383

Sehell J. and Shochet J. (2001) Designing interactive theme park rides. IEEE Computer Graphics and Applications, Vol. 21(4), 11-13

Shilit B. N., Adams N. I. and Want R. (1994). Context-Aware Computing Applications. In Proceedings of the Workshop on Mobile Computing Systems and Applications. IEEE Computer Society, Santa Cruz, CA, 85-90

Shneiderman B. (1998) *Designing the User interface: Strategies for Effective Human-Computer Interaction.* 3rd Edition, Workingham, Addison-Wesley

Shneiderman B. (1980) *Software Psychology: Human Factors in Computer and Information Systems.* Cambridge, MA, Winthrop

Siggraph (2003) 30th International Conference on Computer Graphics and Interactive Techniques. Exhibition of Emerging Technologies. http://www.siggraph.org/s2003/conference/etech

Slater M. and Wilbur S. (1998) A Framework for Immersive Virtual Environments (FIVE): Speculations on the Role of Presence in Virtual Environments. Presence, Vol. 6(6), 603-616

Strauss A. and Corbin J. (1998) *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory.* 2nd Edition. Newbury Park, CA, SAGE Publications

Stuart R. (1996) *The Design of Virtual Environments.* New York, McGraw-Hill

Tamura H., Yamamoto H. and Katayama A. (1999) Steps Toward Seamless Mixed Reality. In Ohata Y. and Tamura H. (eds.) (1999) *Mixed Reality: Merging Real and Virtual Worlds.* Tokyo, Ohmsha Ltd

Tang J., Yankelovich N., Begole B., Van Kleek M., Li F. and Bhalodia J. (2001) ConNexus to Awarenex: Extending awareness to mobile users. In Proceedings of CHI'01, Seattle, WA, USA, ACM

Teasley B., Lund A. and Bennett R. (1996) Interactive Television: A New Challenge for HCI. In Conference Companion on Human Factors in Computing Systems: Common Ground, CHI'96, ACM

Technology Review (2003) 10 Emerging Technologies That Will Change the World. http://www.technologyreview.com /articles/emerging0203.asp

Weiser M. (1991) The Computer of the 21st Century. Scientific American, Vol. 165(3), 66-75

Winograd T. (ed.) (1996) *Bringing Design to Software.* New York, ACM Press

Witmer B. G. and Singer M. J. (1998) Measuring Presence in Virtual Reality: A Presence Questionnaire. Presence, Vol. 7(3), 225-240

Wynekoop J. L. and Conger S. A. (1990) A Review of Computer Aided Software Engineering Research Methods. In Proceedings of the IFIP TC8 WG 8.2 Working Conference on The Information Systems Research Arena of The 90's, Copenhagen, Denmark

Yin R. K. (1994) *Case Study Research, Design and Methods.* 2nd Edition. Newbury Park, Sage Publications

List of Pictures

Appendix A

Paper Contributions

Published in Smith W., Thomas R. and Apperley M. (eds.) (2001) *Proceedings of OzCHI 2001* (Perth, Australia, November 2001). Churchlands, Edith Cowan University Press, pp 77-83

COMBINING INTERACTION TECHNIQUES AND DISPLAY TYPES FOR VIRTUAL REALITY

Jesper Kjeldskov

Department of Computer Science, Aalborg University, Denmark
email: jesper@cs.auc.dk
http://www.cs.auc.dk/~jesper

ABSTRACT

Full and partial immersion in virtual reality are fundamental different user experiences: partial immersion supports the feeling of "looking at" a virtual environment while full immersion supports the feeling of "being in" that environment. Working with a range of interactive virtual reality applications using different display systems we have found that the use of six-sided caves and panoramic displays results in different requirements to interaction techniques. These can be related to specific categories of interaction: orientating, moving and acting. In this paper I present a framework for the evaluation and design of interaction techniques for virtual reality focusing on the relations between interaction techniques and display types.

KEYWORDS

Virtual Reality, Interaction Techniques, Interaction Devices, Display Types, Field of View.

1. INTRODUCTION

A central issue within human-computer interaction is the creation of better ways of interacting with computers (Dix et al. 1998, Preece et al. 1994, Shneiderman 1998). Designing good interaction techniques in new areas of computing such as virtual reality is, however, not trivial. Interaction devices and display types are numerous and general knowledge on the implications of combining the two is needed. How are interaction techniques influenced by the use of different display types? Which parts of the interaction is influenced by which display types and why? How can display types be categorized appropriately in relation to interaction? In this paper I address these questions. I start out by presenting the experiment. I then present an overall categorization of display types for virtual reality and a division of the concept of interaction based on our findings. The categorization of displays and division of interaction are then related in a matrix summarizing our evaluation of interaction techniques.

2. THE EXPERIMENT

A qualitative pilot investigation of interaction in virtual reality was conducted over a period of 6 months. For this investigation we used two different display systems: an 8x3 meter cylindrical panoramic display covering a field of view (fov) of 160° and a six-sided cave measuring 2½ meter on each side. For interaction we used Polhemus Fastrak motion tracking, an SGI Spacemouse (3D mouse), a "magic wand" (3D joystick) and a wireless trackball from Logitech with tracking via the Polhemus system. During the investigation we implemented and evaluated the usability of more than 40 interaction techniques for specific combinations of these displays and interaction devices. The interaction styles were tested within a number of virtual reality applications ranging from industrial design and simulation to interactive games and virtual reality art. The interaction techniques were tested in random order.

Figure 1: The six-sided cave

Figure 2: The panoramic display

Six test users participated in the experiment. The test users were all experienced with virtual reality but had no particular experience with the specific applications. The tasks to be performed by the test users primarily addressed orientating and moving in the virtual 3D spaces of the applications as well as selecting and manipulating 3D objects. The test sessions lasted approx. 10 to 15 minutes pr. interaction technique. Data was collected through qualitative observations of use and statements by the test users noted during and after the test. The test users were asked to describe their experience with the interaction technique: was it hard to learn? Was it hard to operate? Did it cause any problems? Did it limit any part of the interaction? Some statements led to the immediate development and test of further interaction techniques.

3. VIRTUAL REALITY DISPLAYS

The literature on virtual reality indicates use of a wide range of different display types: fishtank virtual reality, head-mounted displays (hmds), boom-mounted displays (booms), holobenches, large panoramic screens and caves with a different number of sides (Schneiderman 1998, Dix et al. 1998, Stuart 1996). These different displays have fundamental different characteristics, potentials and limitations (cf. Buxton et al. 1998, LaViola Jr. 2000).

3.1 Full and partial immersive displays

Full and partial immersion in virtual reality are fundamental different user experiences: partial immersion supports the feeling of "looking at" a virtual environment while full immersion supports the feeling of "being in" that environment (Schneiderman 1998:221-222). The potentials for immersing the user in a virtual environment is often measured from the *field of view* (fov), which describes how much of the user's view can be covered. This suggests that e.g. panoramic displays are more immersive than head-mounted displays. However, as the fov is measured from a fixed position in a fixed direction and users interacting in a virtual environment are typically *not* remaining still, other properties should also be considered.

Figure 3: Full immersive displays for virtual reality: six-sided caves, hmds and booms (hmd mounted on stand and operated by hand)

I suggest the notion of *available field of view* describing the fov available in *any* given viewing direction. If a display always provides an available field of view, it is considered a *full immersive display*. If a display does not always provide an available field of view, it is considered a *partial immersive display*. Using this notion, display types for virtual reality can be categorized as shown in figure 3 and 4.

Though hmds and booms have relatively low fov compared to holobenches and panoramic screens, it is available in all directions the user may be orientating due to the construction of the display. This makes hmds and booms full immersive. The opposite is the case with most large stationary virtual reality installations such as holobenches, powerwalls, panoramic screens and 3-5 sided caves. These display types only provide their (high) fov within a given direction. Of special interest is the fact that caves fall into both categories depending on the availability of the 6th side. Six-sided caves thus surround the user completely and can be characterized as full-immersive displays whereas 3-5 sided caves are only partial immersive.

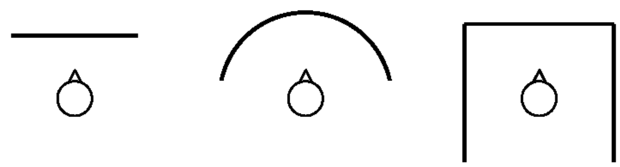


Figure 4: Partial immersive displays for virtual reality: monitors/holobenches, panoramic screens and 3-5 sided caves.

The partial immersive displays depicted in figure 4 are characterized by the fact that they do not provide their optimal fov in all directions. When using partial immersive displays the immersive experience is highly vulnerable to the user changing viewing direction, because the display simply leaves out an area where the virtual environment is not projected. When using full immersive displays this is not a problem.

4. INTERACTION IN VIRTUAL REALITY

The literature on interaction in virtual reality suggests that conceptual frameworks for interaction can influence significantly on the quality of interaction techniques. A characterization of universal interaction tasks in virtual reality and taxonomy for interaction techniques is presented in Bowman (1998) and used in Bowman et al. (1998) to create a highly interactive virtual reality application. The prototype, however, makes solely use of a head mounted display and is not considered in relation to different display types. It could be interesting to see how the techniques perform with other displays. A theoretical framework for analyzing manipulation techniques and a testbed for experimenting with different interaction techniques is presented in Poupyrev (1997) followed by taxonomy of manipulation techniques based on a number of prototypes in Poupyrev et al. (1998). The prototypes tested, however, make exclusively use of head-mounted displays. How different display types could contribute to the taxonomy would be interesting.

4.1 Dividing the concept of interaction

We found that using simply the notion of *interaction* made it hard to describe and clearly differentiate the specific problems we encountered. There may be several reasons for this. First the concept of interaction and interactivity suffer from long-term use as buzzwords. Second virtual reality calls for fundamentally new ways of interacting with computers, supporting the user being present inside a virtual world. We therefore found it suitable to divide the concept of interaction into three more specific categories: *orientating, moving* and *acting.*

Orientating oneself in virtual reality addresses the need for being able to look around in a virtual environment developing a sense of presence. A common approach is rotating the virtual world while the user remains still.

Moving in virtual reality addresses the need for being able to move around in a virtual environment. This is typically done by motion tracking. But because virtual worlds are typically larger than the physical area within which they are explored, alternative solutions are necessary. A common approach is letting the user move the virtual world by e.g. a joystick while remaining still.

Acting in virtual reality covers both the tasks of selecting, moving, rotating and transforming objects in the virtual environment as well as control on a system level. Acting is typically supported by variations of *virtual hand* or *virtual pointer* techniques (Poupyrev et al. 1998). Others go beyond this trying to support "natural" acting in virtual environments by means of gesture recognition (Moeslund 2000).

5. DISPLAY TYPES AND INTERACTION TECHNIQUES

In the table below, the lessons learned from our experiment are organized in relation to display types and interaction techniques. Headtracking with “active zones” and “mapped 1:2” is described in section 5.1.

TABLE 1

LESSONS LEARNED: RELATIONS BETWEEN INTERACTION TECHNIQUES AND DISPLAY TYPES

<table>
<tr><th colspan="2">Interaction technique</th><th>Partial immersive displays
(Panorama)</th><th>Full immersive displays
(Six-sided cave)</th></tr>
<tr><td rowspan="6">Orientating</td><td>Headtracking</td><td>1) Problematic because the user can’t orientate 360° by simply looking in the desired direction.</td><td>2) Very intuitive because the user can orientate simply by looking around as in the real world.</td></tr>
<tr><td>Headtracking with active “zones”</td><td>3) Limits freedom of movement. Conflicts with orientating in the physical world.</td><td>n/a</td></tr>
<tr><td>Headtracking mapped 1:2</td><td>4) Easy to learn and fast in use. However makes it hard to gain feeling of presence.</td><td>n/a</td></tr>
<tr><td>Joystick</td><td>5) Easy to use. Better than trackball for fast/continuous rotations.</td><td rowspan="2">6) Supports viewing the virtual environment from “odd perspectives” as addition to headtracking. Frees the user from moving physically. Trackball supports more precise rotations than joystick.</td></tr>
<tr><td>Trackball</td><td>7) Easy to use. Better than joystick for precise/absolute rotations</td></tr>
<tr><td>Spacemouse</td><td>8) Supports rotating and moving the virtual world at same time. Hard to learn how to operate smoothly.</td><td>9) Can be used with hmds but is not well designed for hand-held use in the cave without additional tracking.</td></tr>
<tr><td rowspan="4">Moving</td><td>Position tracking</td><td colspan="2">10) Very intuitive to use within the limits of the physical space available. However, typically demands further support for moving the virtual world (e.g. by means of joystick, trackball or Spacemouse).</td></tr>
<tr><td>Joystick</td><td colspan="2">11) Flying in the direction the stick is moved. Easy to use but not suited for both fast and precise movements. Need “gears” to control flying speed. Supporting orientating with same device is problematic – demands shifting between several modes of operation.</td></tr>
<tr><td>Trackball</td><td colspan="2">12) Flying in the direction the ball is rolled. Great feeling of control over small movements. Not suited for moving long distances. Supporting orientating with same device is problematic – demands shifting between several modes of operation.</td></tr>
<tr><td>Spacemouse</td><td>13) Works fine if user remains still. Performs well along with headtracking but is hard to learn to operate smoothly.</td><td>14) Does not work well. Device must stay in fixed orientation in relation to the display. Is hard to learn to operate smoothly.</td></tr>
<tr><td rowspan="2">Acting</td><td>Virtual hand (using tracking)</td><td>15) Does not support “close-by” acting unless user stands close to the screen.</td><td>16) Can be projected close to physical hands but the user’s body might occlude graphics.</td></tr>
<tr><td>Virtual pointer (using tracking)</td><td>17) Large screens have good affordances for “pointing at” objects in a VE.</td><td>18) May support moving in pointing direction by indicating direction visually</td></tr>
</table>

From the data summarized in table 1, we identified a number of interesting issues. Due to limited space, however, only four of these issues are described in detail in the following sections.

5.1 Untraditional use of headtracking

We implemented two interaction techniques that rotated the world by the use of headtracking (table 1, issue 1-4). These techniques each had significant downsides. Mapping the horizontal orientation of the headtracker 1:2 to the horizontal rotation of the virtual world facilitated “turning around” in the virtual environment by looking only 90° to the left or right. Though satisfied with the speed ease and speed of use, test users complained that the technique made them disoriented and seasick. Using a technique that rotated the virtual world in a given direction by looking towards the edges of the display (“active zones”), did not result in these problems. Test users, however, complained that they could not face away from the display without spinning the virtual world. A button for deactivating the active zones was suggested. Though having potentials for orientating in partial immersive displays, mapping the user's orientation this way thus challenges the boundary between interacting in the virtual and the physical world.

5.2 Complementing headtracking in full immersive displays

Using full immersive displays, test users expressed a need for viewing the virtual environment from perspectives, which were hard to obtain by moving physically. We therefore developed techniques for rotating the virtual world in the cave by using joystick or trackball while remaining still (table 1, issue 6 and 9). All users expressed satisfaction with this technique as a supplement to headtracking. Observing test users playing CAVEQuake in the six-sided cave, however, revealed that it caused the users to *always* use the joystick when "turning around" rather than turn around physically. Test users at the same time surprisingly expressed, that the level of immersion in the six-sided cave was not much higher than when using the panoramic screen. We then disabled rotation by means of joystick, forcing the user to turn around physically. The immersive effect was enormous. Test users reported a significant higher level of immersion and within few minutes had difficulties defining the position of the cave-walls and identifying which one was the door. Complementing headtracking in full immersive displays by means of joysticks thus may be relevant in some applications but involves a possible loss of immersiveness in others.

5.3 Use of 3D interaction devices

We implemented interaction techniques, which used a Spacemouse providing 6 degrees of freedom. Pushing the top of the Spacemouse caused the user to "fly" in that direction. Twisting it rotated the virtual world (table 1, issue 8, 9, 13 and 14). When seated in front of the panoramic screen with the Spacemouse located on the armrest, test users reported that the technique worked fine – though it demanded some use experience. When using the Spacemouse in a full immersive display, however, the test users reported that the technique was unintuitive. If not holding the device the correct orientation in relation to the display, moving it e.g. forward thus caused the user to fly in a completely different direction. This was reported very confusing. When keeping the Spacemouse in the correct orientation, however, the test users reported that the technique gave them a good feeling of control compared to the use of joysticks and trackballs. Compensating for the orientation of the device by motion tracking was suggested – but not implemented.

5.4 Supporting different techniques for acting

We implemented and tested a virtual hand and a virtual pointer technique (table 1, issue 15-18). Virtual hand techniques provide representations of the user's hands while virtual pointer techniques to a large extend resemble the use of laser pointers. Using the full immersive display, test users reported that the virtual hand approach was very intuitive for "close-by" interaction. Picking up, moving and rotating virtual objects was reported unproblematic. The user's physical hands, however, sometimes occluded the graphics. The test users reported less satisfaction with the virtual hand approach when using a partial immersive display. Due to the lack of a floor display in the panoramic display setup, the virtual hands could not be projected close to the user's hands unless standing very close to the display. The virtual pointer was reported usable in combination with both the panoramic display and the cave when having to point at or pick something in the virtual environment. Moving and rotating objects with the virtual pointer was, however reported problematic in both the partial and the full immersive display.

6. CONCLUSIONS

The primary conclusion from our experiments is that the same interaction techniques does not work equally well with panoramic displays and six-sided caves. Displays for virtual reality can be categorized as full or partial immersive depending on their *available field of view*. Using this categorization in relation to a division of interaction into *orienting, moving* and *acting* reveals a series of issues for the design of human-computer interaction in virtual reality applications. We specifically found that:

1. Untraditional use of headtracking may support orientating in partial immersive displays, though introducing a problematic boundary between interaction in physical and virtual space.

2. Rotating the world in full immersive displays using e.g. joysticks may complement headtracking by letting the user view the VE from odd perspectives but might limit the feeling of immersiveness.

3. Non-tracked 3D interaction devices work fine for orientating and moving in partial immersive displays but can be problematic in full immersive displays if the user turns around physically.

4. The virtual hand approach has different potentials in partial and full immersive displays. The virtual pointer approach works fine for picking 3D objects but is problematic for rotating and moving them.

For new and better ways of interacting in virtual reality to emerge, system developers must optimize combinations of devices/techniques and displays in specific application contexts. The framework presented in this paper may support a structured approach to this task. The presented framework should be developed further by structured and quantitative explorations of the relations between interaction techniques and display types.

7. ACKNOWLEDGEMENTS

I wish to thank Peer M. Ilsøe for implementation, Lars Qvortrup, The Staging Project and VR-MediaLab for funding and access to VR systems. I also thank the test users: Peer M. Ilsøe, Tom Nyvang, Thomas D. Nielsen, Carsten B. Larsen, Gabriel S. Hansen and Claus A. F. Rosenstand. I thank Jan Stage and Mikael B. Skov for reviewing the paper and Erik Granum for inspiration and constructive comments on my work.

8. REFERENCES

Bowman, Doug A. (1998). Interaction Techniques for immersive Environments: Design, Evaluation and Application. *Human-Computer Interaction Consortium (HCIC) Conference, 1998.*

Bowman Doug A. et al. (1998). The Virtual Venue: User-Computer Interaction in Information-Rich Virtual Environments. *Teleoperators and Virtual Environments,* **7: 5,** 478-493.

Bowman, Doug A. et al (2000). 3D User Interface Design: Fundamental Techniques, Theory and Practice. *Course Notes no 36, Siggraph, New Orleans, 23-28 July 2000.*

Buxton, Bill and Fitzmaurice, George W. (1998). HMD's, caves & Chameleon: A Human-Centric Analysis of Interaction in Virtual Space. *Computer Graphics, The Siggraph Quarterly,* **32: 4,** 64-68

Dix, Alan (ed.) et al. (1998). *Human-Computer Interaction – Second Edition,* London, Prentice Hall.

LaViola Jr., Joseph J. (2000). Input and Output Devices. *Course notes in Bowman et al. (2000).*

Moeslund, Thomas (2000). Interacting with a Virtual World through Motion Capture. *In Qvortrup (2000).*

Poupyrev, Ivan et al (1997). A Framework and Testbed for Studying Manipulation Techniques for Immersive VR. *ACM VRST 1997, Lausanne, Switzerland; New York, ACM, 21-28.*

Poupyrev, Ivan et al (1998). Egocentric Object Manipulation in Virtual Environments: Empirical Evaluation of Interaction Techniques. *Eurographics'98,* **17: 3,** Oxford, Blackwell Publishers.

Preece, Jenny, Y. Rogers, H. Sharp, D. Benyon (1994). *Human-Computer Interaction,* Workingham, Addison Wesley.

Qvortrup, Lars (ed.) (2000). *Virtual Interaction: Interaction in Virtual Inhabited 3D Worlds,* London, Springer-Verlag.

Shneiderman, Ben (1998). *Designing the User Interface: Strategies for Effective Human-Computer Interaction, 3rd Edition,* Reading, MA., Addison Wesley.

Stuart, Rory (1996). *The Design of Virtual Environments,* New York, McGraw-Hill.

Extended version of Kjeldskov J. and Kolbe N. (2002) Interaction Design for Handheld Computers. Published in Dai G., Dong S., Chan Y. and Ren X. (2002) *Proceedings of the 5th Asia Pacific Conference on Computer Human Interaction, APCHI 2002* (Beijing, China, November 2002). Beijing, Science Press, pp 433-446

Combining Input Devices and Interaction Styles for Handheld Computers

Jesper Kjeldskov and Nikolaj Kolbe

Aalborg University, Department of Computer Science
Fredriks Bajers Vej 7E, 9220 Aalborg East, Denmark
jesper@cs.auc.dk

ABSTRACT

Interaction design for handheld computers challenges our basic knowledge about human-computer interaction. Displays are small due to limited physical size of the devices and interaction is limited due to handheld operation. While a lot of effort is being put into the development of new means of input and the design of miniature graphical interfaces, little research is reported on the usability of the variety of already available input devices for handheld computers in relation to common styles of interaction being used. Reporting from an empirical usability experiment, this paper addresses the relations between input devices commonly available on mobile devices today and classical styles of interaction described in HCI research literature. The question asked is simple: how do which input devices fit which interaction styles?

Keywords: Handheld computers, mobile interaction, input devices, interaction styles, usability evaluation

1. INTRODUCTION

Handheld computers take various forms supporting both general and highly specialized use. Personal digital assistants provide calendar-like functionality and applications useful when traveling such as simple tools for communication and information retrieval, miniature office applications and applications for leisure and entertainment. In the more specialized end, mobile phones primarily support spoken and written communication but typically also provide simple applications for managing the user's address book etc. Some mobile phones furthermore support simple Internet browsing. Other handheld computers have even more dedicated functionality such as e.g. GPS positioning and route planning devices, MP3 players, infrared bar code readers or pocket calculators. Though diverse, these devices share the problem of interaction design.

Squeezing complex functionality into the interface of a pocketsize computer and putting it in the hand of a mobile user represents a serious interface and interaction design challenge. Displays are small due to limited physical size of the devices and interaction is limited due to handheld operation. This challenge requires additional knowledge about human-computer interaction in a mobile use context to be obtained.

The purpose of this paper is to discuss the relevance and compare the usability of *traditional* input devices and styles of interaction for handheld computers. We specifically compare the usability of three different input devices in relation to three different styles of interaction, providing a list of properties describing the relations between the two being useful when matching interaction styles to available input devices and visa versa.

The paper is structured in the following way. In the next section, we approach the issue of interaction design for handheld computers, stressing the relevance of extended knowledge about the usability of traditional means of interaction in this context. In section three, we line up a number of commonly available input devices for handheld computers today, describing their immediate properties and limitations. In section four, we then turn our focus towards the use of traditional interaction styles such as menu selection, form filling and direct manipulation in the context of handheld computers.

The comparison of input device and interaction style relations are based on an empirical usability experiment. In section five this experiment is described, followed by an outline of our findings in section six. Finally we conclude on the discussion and point out limitations and avenues for further work.

2. INTERACTION DESIGN FOR HANDHELD COMPUTERS

Interaction design for handheld computers is approached from various perspectives including the development of new technologies as well as creative exploration of existing ones. A lot of effort is specifically being put into the development of new means of input such as speech [1], environmental sensors [2] and context awareness [3] etc. In parallel, a lot of research is done within the design of miniature graphical interfaces [4], the exploration of multi-modal output [5] and the use of alternative display technologies e.g. build into ones eyeglasses [6][7].

Studying the research literature on mobile HCI one thus quickly gets the impression that interaction with mobile computers could soon be fundamentally different from traditional computer use.

Nevertheless, most commercially available handheld computers today are still dominated by the use of rather traditional means of interaction design known from the human-computer interfaces of graphical desktop computers. For a quick overview of the design of mobile device interfaces see e.g. [4]. The interactions design encouraged by general handheld operative systems such as PocketPC, PalmOS, EPOC and Symbian thus to a large extend resembles that of their desktop counterparts in a miniature scale.

While the use of human-computer interaction techniques originally developed for desktop computers may without dispute be inconvenient in *some* mobile use contexts, this approach yet seems to have a huge justification. The immediate advantages are obvious. People are used to interacting with computers through windows, icons, menus, pointers and function keys throughout years of experience with personal and office computing. Applying the same approach to interaction design with handheld computers, users can benefit from well-known conventions of design and use. Similarly, designers are used to express and communicate the functionality of systems through such interfaces throughout years of system development practice.

Designing and using fundamentally *new* kinds of interfaces for handheld computers thus involves both users and designers to throw out some of their existing knowledge and start over again: develop and assimilate new conventions of human-computer interaction. If not gaining immediate significant quantitative and qualitative improvements of technology use such as increased effectiveness, lower rates of error and higher subjective satisfaction, the effort of designing and learning new interaction design may not be worth the while from neither a developer nor a user perspective: if it pretty much works as it is… why change it into something that might not be better? In this perspective, however, knowledge about traditional means of interaction in the context of mobile computing is important.

2.1. Input Devices and Interaction Styles

The usability of any interaction design depends on the relation between *interaction style* and *input device.* Whereas input devices are typically physical artifacts like keyboards, pointers, joysticks, microphones, cameras etc., interaction styles represent ways by which the specific input from these devices is translated and used by the computer application such as e.g. direct manipulation, command language or menu selection.

The HCI literature often presents the choice of input devices as a matter of suiting the desired interaction style best [8][9][10]. Alphanumeric keyboards are chosen for text entry. Mice are chosen for direct manipulation and virtual reality gloves are chosen for three-dimensional interaction. Within the context of handheld computing, however, one may argue that this relation is reversed because the available means of input are typically limited.

Despite a huge diversity of functionality and use, most handheld computers are characterized by facilitating the same input devices. While the use of alphanumeric keyboards and indirect-pointing devices; mice, trackballs and joysticks is limited within the context of handheld computing due to physical size and lack of support for mobile operation, numerical keyboards, direct-control pointing devices (such as pens and touch screens), cursor-movement keys and function keys are highly prevalent. Numerical keyboards are used for entering numbers and text on mobile phones. Direct pointer are used for selecting and manipulating elements on graphical displays or for recognizing handwriting. Cursor movement keys/scroll wheels are used for browsing lists of information such as one's phone book and function keys are used for activating dedicated functions such as answering a call or starting the voice recorder application.

The usability of these input devices have been thoroughly evaluated in relation to desktop computing and documented in the HCI literature (e.g. [8][9][10]). However little similar research has been reported on the usability of the different input devices typically facilitated on handheld computers in relation to the traditional styles of interaction, which is often applied [4]. Limited knowledge thus exists on how these interaction styles perform in relation to the available input devices for handheld computers. The question is simple: how does which input devices fit which interaction styles?

3. INPUT DEVICES FOR HANDHELD COMPUTERS

In the context of handheld computing, numerical keyboards, cursor-movement keys, function keys and touch screens share the property of being relatively small and naturally embedded into the devices controlled by them. Furthermore, most of these input devices can potentially be operated with the same hand that is also holding the device.

Function keys on handheld computers specifically have the advantage of supporting quick access to frequently used applications or functions. Function keys are often labeled with icons or text but sometimes simply signal their functionality through their shape and/or location on the device, known as natural mapping. The use of function keys is normally restricted by the limited number of functions potentially assigned to each of them. A common solution to this problem is to have different functions assigned at different times according to the state of the device or application (known as soft keys).

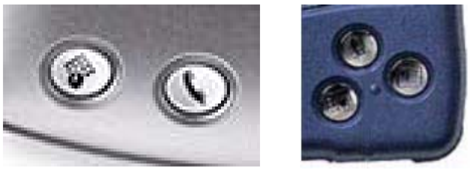


Figure 1. Function keys on handheld computers

Cursor movement keys on handheld computers are typically used as a more general purpose input device than function keys. Cursor keys embedded into handheld computers have the advantage of being naturally mapped to the physical layout of the device controlled by them. Cursor keys are typically either shaped as or labeled with arrows or grouped into multidirectional buttons, as illustrated below. While function keys have affordances for discrete input (single clicks), cursor keys are typically *also* used for continuous input (click and hold). The use of cursor keys is restricted by the limited speed of use compared to e.g. direct or indirect pointing devices.

Figure 2. Cursor keys on handheld computers

Direct pointers for handheld computers exploit the fact that a graphical display is typically embedded into the device. Pointing directly on the screen using ones finger or by means of a pen are obvious affordances of such design. Direct pointers are primarily limited by the demand for two-handed operation, the need for a separate pointing device, the lack of precision while moving and the visual occlusion of the display while pointing. Like cursor keys, direct pointers have affordances for both discrete input (selecting) *and* continuous input (dragging/writing).

Figure 3. Direct pointers on handheld computers

4. INTERACTION STYLES FOR HANDHELD COMPUTERS

While available human computer interaction styles are numerous [8][9][10], only a few styles of interaction are being widely used within the area of handheld computing. Performing a quick survey into a broad range of applications commercially available for handheld computers, three styles of interaction are clearly dominating the picture:

- Menu selection
- Form filling
- Direct manipulation

Figure 4. Menu selection, form filling and direct manipulation in graphical interfaces for the PalmPilot

Menu selection is widely used in handheld computer interfaces. Menu selection has the advantage of supporting the representation of a large number of possible actions in a relatively limited space using only simple means such as written text and small icons. This is essential when designing for small displays with limited graphical performance. Furthermore, interaction design based on menu selection is highly versatile. Selecting an item or action from a menu can be done in various ways: e.g. by entering a corresponding number for each line or selecting a specific line either by browsing the list or pointing it out. This versatility facilitates the use of various input devices: numeric keyboards, cursor keys and pointers.

Form filling is also very common in interfaces for handheld computers. Contrary to menu selection, form filling is used for entering more complex input where a list of all possible choices cannot be provided or would be too comprehensive, such as names, addresses or phone numbers. Form filling has the advantage of guiding the user through the interaction and structuring user-input by explicitly indicating what information is required at a specific time and where to put it. Form filling demands interaction on two levels. First, the field to be filled has to be selected. Secondly, input has to be entered (either text or numbers). While fields of a form can be selected much like the elements in a menu, entering text and numbers into the fields is often problematic on a handheld computer, limiting the use of this interaction style. The suggested solutions are numerous depending on available input devices: from handwriting recognition and word prediction, to picking letters on a virtual keyboard using a pen or browsing through the alphabet using a set of dedicated keys.

Direct manipulation interfaces in which the user selects and manipulates virtual objects on the screen are prevalent in handheld computer interfaces but not as commonly used as menus and forms. Although some applications allow the user to pick and manipulate virtual objects directly on the screen, direct manipulation on handheld computers is primarily used for e.g. drawing, selecting and moving text and for operating menus and forms.

According to Shneiderman [8], when it is not possible to create appropriate direct-manipulation interfaces, menu selection and form filling are good alternatives.

5. THE EXPERIMENT

An experiment was conducted evaluating the usability of the three described input devices for handheld computers in relation to the three described styles of interaction. This section explains the design of that experiment.

The experiment was divided into two phases. First, a series of experimental prototypes were designed and implemented for handheld computer use with the scope of comparing the relations between input devices and interaction styles discussed above. Secondly, a usability evaluation of these prototypes was carried out. A development and evaluation team of three people designed and implemented the prototypes and conducted the usability evaluation.

The experiment was inspired by earlier work evaluating the usability of different input devices in relation to *one* style of interaction [11] and evaluating the usability of a number of different interaction styles in combination with *one* specific input device [12]. While these experiments, however, varied only one parameter (either input device or interaction style), we decided to set up at matrix of the two, allowing us to evaluate several combinations.

5.1. Experimental prototypes

Three different prototypes of the same information service facilitating simple browsing of movies and reservation of seats in a fictive cinema were implemented for use on a handheld computer.

Each prototype of the information service required the use of form filling, menu selection and direct manipulation but was designed for interaction through only one of the three input devices: function keys, cursor keys or direct pointer.

The prototypes were based on analysis and design conducted using the ETP method [13] for object modeling and user interface design. The implementation was done using Microsoft embedded Visual Basic 3.0 and a software development kit for PocketPC.

The following illustrations show the interface design for interaction using a direct pointer.

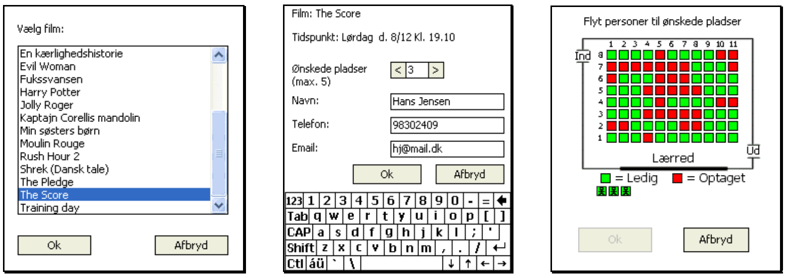


Figure 5. Menu selection, form filling and direct manipulation for experimental cinema information service

1. Selecting a movie from the list
2. Entering number of seats and contact information
3. Specifying the desired vacant seats

5.1.1. Selecting a movie

On the first screen, the user is requested to select a movie from a menu. Using function keys, the list can be divided into two (upper or lower half) until only one item remains. Using Cursor keys, the user can scroll up and down until the desired item is highlighted. Using a direct pointer, the user can select an item by clicking on it.

5.1.2. Entering contact information

On the second screen, the user is requested to enter number of seats and fill in contact information (name, phone number and e-mail address) into a simple form. Using dedicated function keys, the user can browse through the alphabet letter by letter, acknowledge a letter and jump to the next field. Using the five way-cursor key, the user can similarly browse the alphabet (left/right), acknowledge a letter (middle) or jump to the previous or next field (up/down). Using the direct pointer, the user can point at a specific field and select each letter one at a time from a virtual keyboard on the screen.

5.1.3. Specifying seats

On the third screen, the user is required to specify the desired seats by moving the icon of a small group of people (corresponding to the selected number of seats) to the preferred vacant location in the cinema by means of direct manipulation. Using function keys, the group of persons can be moved one step left or one row down. When reaching the desired location, this can be acknowledged by pressing a third dedicated button. Using cursor movement keys, the group can be moved up/down or left/right. Location is acknowledged by pressing the center of the button. Using the direct pointer, the group of people can be dragged to the desired location, which is then acknowledged by clicking the “ok” button.

5.1.4. Hardware

The prototypes were targeted at a Compaq Ipaq 3630 handheld computer with 32MB RAM and a color display of 240x320 pixels running Microsoft PocketPC. This device facilitated interaction by means of four function keys and a five-way cursor movement button located below the display, or by touching the display using either ones finger or a dedicated pen as depicted on figure 6.

Figure 6. Function keys, cursor keys and direct pointer on Ipaq

5.2. Usability evaluation

An evaluation of the interaction design implemented in the experimental prototypes described above was carried out. The evaluation was conducted at a dedicated usability laboratory at the University of Aalborg, DK.

5.2.1. Participants

A total of 21 male and female students of computer science or informatics at the University of Aalborg participated in the usability evaluation. The test subjects were equally distributed among the three prototypes according to their study orientation, sex and experience with the use of PDAs. Each test subject was assigned to one of the three input devices and used this in combination with all three styles of interaction. Seven users thus tested each input device.

5.2.2. Experimental procedure

The usability evaluation was divided into two phases. First, each user was given a brief lesson in the principle and use of the input device and interaction styles to be evaluated. The prototypes supported this through a simple tutorial application. Hereafter, the user carried out an overall task, using the dedicated input device.

5.2.3. Tasks

The overall task to be carried out by the test subjects consisted of three more specific tasks corresponding to the scope of the prototypes. 1) Selecting a specific movie from a list using menu selection. 2) Specifying number of seats and entering personal information using form filling. 3) Specifying the desired seats using direct manipulation

5.2.4. Data collection

Time spend on each task was automatically registered through a time log module build into the prototypes, generating a log file following each evaluation. User interaction with the handheld computer was furthermore recorded on audio/video tape and notes regarding the user interaction or comments were taken during the evaluation.

5.2.5. Data analysis

The primary dependant variables of the experiment were task completion times for each of the nine combinations of input devices and interaction styles. Based on the task completion times, we have calculated and compared total and average completion times as well as standard deviations for input devices, interaction styles and users.

5.2.6. Limitations

As the three interaction styles applied in the prototypes were assigned to the solution of different and incomparable tasks (see 5.2.3.) the time spent using each of the interaction styles cannot be compared directly.

6. FINDINGS

In this section, we present and discuss the findings from the experiment. The evaluation shows that direct pointing devices are very versatile and can be successfully combined with any of the three interaction styles. Cursor keys also prove useful in combination with menu selection and direct manipulation while less successful for form filling. Function keys generally have the lowest performance.

6.1. Task Completion Time

The following figures 7-9 show an overall view of measured task completion time for test subjects grouped in accordance to interaction styles and input devices.

Figure 7 shows a variation of task completion times for form filling with reference to the interaction devices used. Direct pointers are clearly in favor. A variation among the test subjects of each interaction device for form filling is also indicated with ranges of task completion times for function keys, cursor keys and direct pointer of 72, 60 and 22 respectively. Again, direct pointer is the favored device with least dispersion.

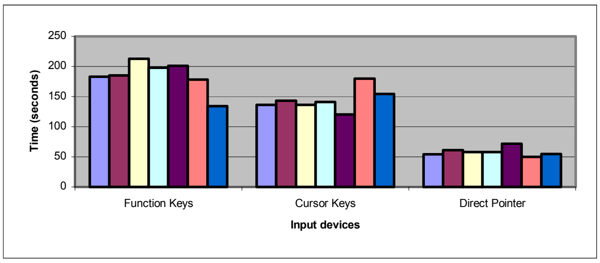


Figure 7. Completion times for form filling

Figure 8 identifies a similar variation of task completion times for menu selection in relation to the interaction devices used. Function keys are clearly being more time demanding than cursor keys and direct pointer. A variation among the test subjects is also indicated. With ranges for function keys, cursor keys and direct pointer of 18, 11 and 7 respectively, the dispersion of task completion time is however considerably smaller than for form filling.

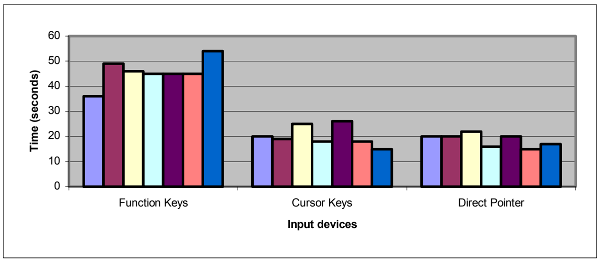


Figure 8. Completion times for menu selection

In figure 9 the variation of task completion times for direct manipulation in relation to the interaction devices used is considerably smaller than for form filling and menu selection. Again, function keys demand more time than cursor keys and direct pointer, but the differences are not as significant as in combination with form filling or menu selection. The variation within the performance of the test subjects is, however, a bit higher than that of menu selection with ranges of 25, 14 and 14 for function keys, cursor keys and direct pointer respectively

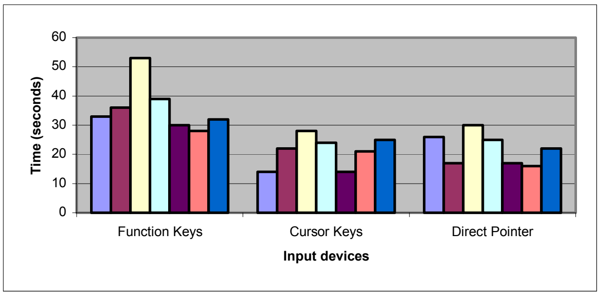


Figure 9. Completion times for direct manipulation

6.2. Total and Average Completion Time

In the following, focus is turned towards discussing and comparing the *effectiveness* of the evaluated input devices. This is done on basis of the total and average task completion time.

Table 1 shows the total and average task completion times for the nine combinations of input devices and interaction styles. These values are also depicted in figure 10 and 11. To the right, the standard deviation of the time spent on each interaction style using the three input devices is calculated.

	Function Keys	Cursor Keys	Direct Pointer	Standard deviation
Form Filling	185	144	58	52,9
Menu Selection	46	20	19	12,5
Direct Manipulation	36	21	22	6,8
Total	267	185	99	

Table 1. Average task completion times (seconds)

Figure 10 illustrates the summed average task completion time using function keys, cursor keys and direct pointer. From this figure it is clear that the use of direct pointer results in a considerably higher performance for solving the same task. The use of a direct pointer thus demanded only 37% of the time spent on function keys and only 54% of the time spent on cursor keys.

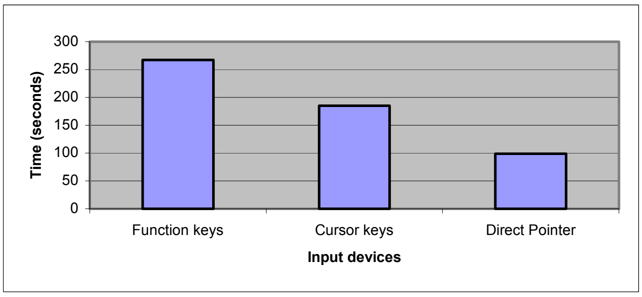


Figure 10. Total average times spent using each input device

Figure 11 shows the distribution of average task completion times for form filling, menu selection and direct manipulation using the three different input devices.

This figure shows that the primary difference in average task completion time originates in the form-filling task. The figure also shows that cursor keys and direct pointers are more or less equally suited for menu selection and direct manipulation. When it comes to form filling, however, the use of a direct pointer demanded only 40% of the time spent using cursor keys and only 31% of the time spent using function keys.

Function keys clearly have the lowest performance in combination with all styles of interaction. Menu selection with function keys consumed 130% more time than when using cursor keys or a direct pointer. Direct manipulation with function keys consumed 70% more time.

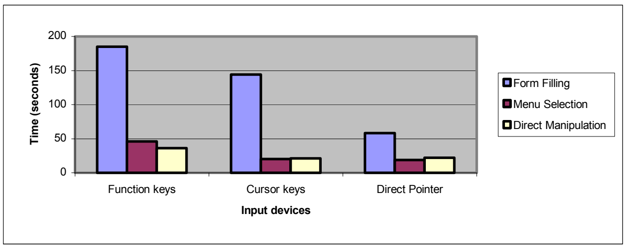


Figure 11. Average task completion times

6.3. Standard Deviations

Having discussed effectiveness, Focus is now turned towards discussing and comparing the *homogeneity* of the interaction styles evaluated and the user performance registered. This is done on basis of the standard deviations of the task completion times for the different input devices, interaction styles and test subjects.

The standard deviations in table 1 represent the level of dispersions of average task completion times for a specific interaction style used in combination with the three different input devices. These are depicted in figure 12. The dissimilar values indicate an inhomogeneous relation between input devices and interaction styles. Compared to the other two styles of interaction, the standard deviation for form filling is rather high (52,9). This implies that dispersions of task completion times are relatively large and that the performance of form filling highly depends on the associated input device. Menu selection, on the other hand, is much less sensitive to the specific input device being used (12,5) while the performance of direct manipulation interaction styles varies least in combination with the different input devices (6,8).

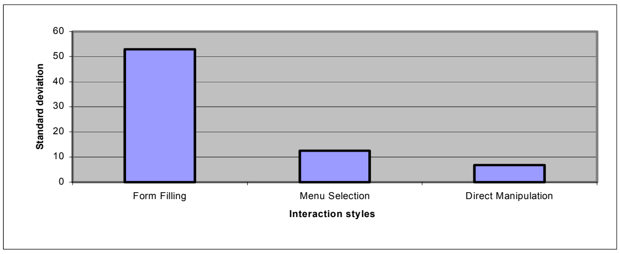


Figure 12. Standard deviations for average completion times

Table 2 show the standard deviations of the measured task completion times for the different test subjects in relation to a specific combination of input device and interaction style. These are depicted in figure 13. The values can be compared in both dimensions representing the user interaction homogeneity in relation to either input device or interaction style.

The standard deviations of these values show that users perform most homogenously by menu selection or direct manipulation interaction styles (1,1 and 1,3) and direct pointer input devices (1,7). Conversely, users perform least homogenously by form filling (7,1) and function keys (8,1).

	Function Keys	Cursor Keys	Direct Pointer	Standard deviation
Form Filling	23,5	17,3	6,4	7,1
Menu Selection	5,0	3,7	2,4	1,1
Direct Manipulation	7,8	5,0	5,0	1,3
Standard deviation	8,1	6,1	1,7	

Table 2. Standard deviations for input devices and interaction styles

The dispersions of standard deviations illustrated in figure 13 show that form filling by the use of function keys or cursor keys causes inhomogeneous and more dispersed user performance relatively to the other styles of interaction. Using menu selection or direct manipulation in combination with any of the three input devices, user performance is, however, highly homogenous.

Figure 13 also show that the use of function and cursor keys generally results in higher standard deviations among users relatively to direct pointers. This indicates that users perform less homogenously when using function and cursor keys, regardless of the applied interaction style.

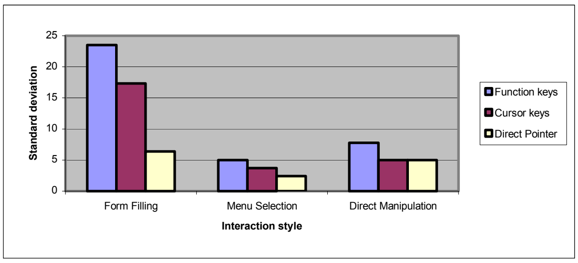


Figure 13. Standard deviations for interaction styles

7. CONCLUSIONS AND DISCUSSION

The combination of input devices and interaction styles for handheld computers has influence on the effectiveness and consistency of user performance. The conducted experiment indicates the following specific conclusions:

Direct pointers have the best overall performance in respect to being both effective and working equally well in combination with all three styles of interaction. Furthermore, users perform most homogenously when using a direct pointer compared to other input devices.

Cursor keys perform as well as direct pointers for menu selection and direct manipulation but are less effective for form filling.

Function keys perform significantly poorer than the other input devices regardless of interaction styles and causes inhomogeneous user performance.

Menu selection supports the most homogenous user performance compared to the other styles of interaction, and performs homogenously with all input devices.

Direct manipulation performs equally effective and equally stable in combination with cursor keys and direct pointers. User performance is homogenous.

Form filling is highly sensitive to the input device being used, with a clear preference for the use of direct pointer, and causes users to perform least homogenously.

These conclusions are discussed more detailed in the following subsections.

7.1. Level of Versatility and Level of Required Input

The conclusions highlight a number of interesting issues. Firstly, it is noticeable that input devices for handheld computers perform differently (table 1 and 2). As outlined above, direct pointers are fastest to use and result in most homogeneous user performance. Cursor keys perform slightly poorer and are less homogenous. Function keys are significantly slower that the other two and causes very inhomogeneous user performance. There may be several reasons for this diversity, but one explanation can be found by categorizing input devices by their restriction on interaction. For this we propose the notion *level of versatility*. Providing only discrete input and being mapped to specific functions, function keys are characterized by low level of versatility. Providing discrete and continuous input mapped to a cursor moveable in steps, cursor keys are characterized by medium level of versatility. Providing discrete and continuous input mapped to a freely moveable pointer, direct pointers are characterized by high level of versatility. This is outlined in table 3 below.

	Input	Use	Level of versatility
Function Keys	Discrete	Mapped to specific functions	Low
Cursor Keys	Discrete, continuous	Mapped to a moveable cursor	Medium
Direct Pointer	Discrete, continuous	Mapped to a moveable pointer	High

Table 3. Classification of input devices: level of constraints

Secondly, interaction styles for handheld computers perform differently too (figure 12 and 13). As described above, the use of direct manipulation and menu selection both result in homogeneous user performance while the performance of form filling results is heterogeneous. Again there may be several reasons for this but one explanation can be found by categorizing interaction styles in relation to the input necessary to operate them. For this we propose the notion *level of required input*. Necessitating browsing, selection of data fields and entering of text and/or numbers, form filling is characterized by a high level of required input. Involving only selection and e.g. drag-and-drop operations, menu selection and direct manipulation is characterized by low level of required input. This is outlined in table 4 below.

	Input	Use	Level of required input
Form filling	Browse, select and entering data	Entering text and numbers	High
Menu selection	Browse and select	Selecting list items	Low
Direct manipulation	Drag-and-drop	Picking and moving icons	Low

Table 4. Classification of interaction styles: level of required input

Collectively, the classifications in table 3 and 4 indicate and explain viable as well as problematic combinations of input devices and interaction styles for handheld computers. Combining an input device with high or medium level of versatility with an interaction style with low level of required input is non-problematic. However, using an interaction style with high level of required input also necessitates an input device with high level of versatility. Likewise, an input device with low level of versatility also requires an interaction style with low level of required input. Combining an input device characterized by low level of versatility with an interaction style characterized by high level of required input will result in poor performance and should be avoided.

7.2. Limited Interaction Space

The final and perhaps most key finding of the study is that cursor keys and direct pointers perform equally well in relation to menu selection and direct manipulation on handheld computers both in terms of efficiency and homogeneity. This is an interesting finding because it contrasts interaction design for traditional desktop computers. Thus, on a large screen, selecting menu items and doing direct manipulation operations using cursor keys would typically not be preferred over the mouse. This indicates that the limited screen size of handheld computers reduces the need for a highly versatile input device. Explaining this phenomenon we propose the notion *limited interaction space* and suggest that for a technology with limited interaction space, complexity is reduced to a degree where input devices with lower level of versatility than direct pointers are sufficient for interaction styles with low level of required input.

Figure 14. Nokia 7650 with five-way cursor key and Ericsson R380s with direct pointer

Based on this it can be discussed, for example, if smart phones with limited interaction space (like the Ericsson R380s) benefit from its pen based touch screen interface or if the same performance would be achieved by means of a five-way cursor key (as on the Nokia 7650), which at the same time also facilitate one-handed operation.

8. LIMITATIONS AND FURTHER WORK

The scope of the work presented in this paper is naturally limited by the interaction design of the experimental prototypes. The applied interaction styles might thus have been implemented differently in order to support the evaluated input devices better. This may have improved the performance of some of the input device/interaction style relations. Furthermore, the experiment is limited in the sense that the performance of the three interaction styles cannot be compared directly. Designing the experiment differently could eliminate this problem.

Further research should test the general value of the presented findings and compare the performance of various interaction styles for handheld computers in relation to different kinds of interaction tasks. Also, detailed comparisons with similar combinations of input devices and interaction styles for desktop computers should be conducted to provide more data on the differences between interaction designs for the two technologies.

ACKNOWLEDGEMENTS

The authors thank the test subjects who participated in the usability evaluation and Morten K. Christiansen and Elena Beck, with whom the experiment was conducted.

REFERENCES

1. N. Sawhney and C. Schmandt, “Nomadic Radio: Speech and Audio Interaction for Contextual Messaging in Nomadic Environments” in ACM Transactions on CHI, vol 7:3, pp. 353-383, 2000.
2. K. Hinckley, J. Pierce, M. Sinclair and E. Horvitz, “Sensing Techniques for Mobile Interaction”, CHI Letters 2(2), pp. 91-100, ACM 2000.
3. K. Cheverst, K. Mitchell and N. Davis, “Investigating Context-aware Information Push vs. Information Pull to tourists” in Proceedings of MobileHCI’01 workshop on HCI with Mobile Devices, IHM-HCI, Lille, France, 2000.
4. E. Bergman (ed.), *Information Appliances and Beyond: Interaction Design For Consumer Products,* London, Morgan Kaufmann, 2000.
5. S. Holland and D. R. Morse, “Audio GPS: spatial audio in a minimal attention interface” in Proceedings of MobileHCI’01 workshop on HCI with Mobile Devices, IHM-HCI, Lille, France, 2000.

6. S. Feiner, B. MacIntyre and T. Höllerer, “Wearing It Out: First steps Toward Mobile Augmented Reality” in Y. Onto and H. Tamura (eds.) *Mixed Reality – Merging Real and Virtual Worlds,* Berlin, Spinger-Verlag, 1999.
7. J. Kjeldskov, “Lessons From Being There: Interface Design For Mobile Augmented Reality” to appear in L. Qvortrup and P. B. Andersen (eds.) *Virtual Applications: Applications With Virtual Inhabited 3D Worlds,* Berlin, Springer-Verlag, 2003.
8. B. Shneiderman, *Designing the User Interface: Strategies for Effective Human-Computer Interaction, 3rd Edition,* Reading, MA, Addison-Wesley, 1998.
9. A. Dix, J. Finlay, G. Abowd and R. Beale, *Human-Computer Interaction – Second Edition,* London, Prentice-Hall Europe, 1998.
10. J. Preece, Y. Rogers, H. Sharp, D. Benyon, S. Holland and T. Carey, *Human-Computer Interaction,* Workingham, Addison-Wesley, 1994.
11. I. S. MacKenzie, A. Sellen and W. Buxton, “A Comparison of Input Devices in Elemental Pointing and Dragging Tasks”, in Proceedings of CHI’91, pp. 161-166, New York, ACM, 1991.
12. I. Benbasat, K. H. Lim and P. A. Todd, “An Experimental Investigation of the Interactive Effects of Interface Style, Instructions and Task Familiarity on User Performance”, in ACM Transactions on CHI, vol 3:1, pp. 1-37, 1996.
13. M. V. Harmelen, *Object Modeling and User Interface Design: Designing Interactive Systems,* Workingham, Addison-Wesley, 2001.

Published in Andersen P. B. and Qvortrup L. (eds.) (2003) *Virtual Applications: Applications With Virtual Inhabited 3D Worlds.* Berlin, Springer-Verlag, pp 159-188

Lessons From Being There: Interface Design for Mobile Augmented Reality

Jesper Kjeldskov
Department of Computer Science
Aalborg University
Denmark
jesper@cs.auc.dk

Introduction

Virtual 3D worlds used to be accessible primarily through display systems separating the users from the physical world and immersing them into a parallel computer-generated virtual reality. Today, virtual 3D worlds are being merged into physical space creating not only *virtual* but *augmented* realities (AR). Contrary to virtual reality, augmented reality facilitates mobility in the real world as well as a close relation between physical space and virtual objects. This makes AR an interesting approach for human-computer interfaces in mobile use-contexts.

Based on the lessons learned from developing “Being There” – an experimental prototype for distributed cooperative work – this chapter addresses the design of augmented reality interfaces for mobile computing devices. Augmented reality is discussed in relation to the concept of *context awareness* in mobile computing and is presented as an interface approach, in which designers must take into consideration the role of physical space on a very high level of contextual detail. One central question is guiding the discussion: how can the properties and use of physical space be analysed and described in a form supporting the creative design of new and fundamentally different human-computer interfaces such as augmented reality? Addressing this question, it is shown how the role of physical space in cooperative work can be formulated in Alexandrian design patterns specifying requirements for human-computer interfaces in a form that identifies desired qualities of an interface, without dictating specific design. From the implemented prototype, it is exemplified how design patterns as interface requirement specifications can be converted into creative interface design for augmented reality.

1. Interface Design for Mobile Devices

The use of mobile computing devices imposes several challenges on the design of human-computer interfaces.

Providing highly relevant input to designers and developers within the mobile device industry here and now, much research on interface design for mobile devices is focused towards exploiting the potentials of existing technology better. This means coming up with new solutions, which overcome the obvious limitations of present mobile technology: small displays, low network bandwidth and limited means of interaction. A number of approaches to these challenges are being explored such as the use of spoken interaction for reducing visual attention and tactile contact with the physical devices, and guidelines for graphical interface design being reconsidered in the context of small screens on mobile devices (see e.g. Bergman 2001). The latter sometimes results in new design more or less contradicting the recognized trends within interfaces for desktop computers:

quality and resolution of images and graphical information representations for use in mobile device interfaces are being *reduced* to the limit of losing their semantic value and information is being squeezed into displays by *removing redundancy* and sometimes breaking it up into a *more complex* structure or hierarchy of sub-sections. Existing information content (on e.g. web pages) is being *filtered* and cut into smaller pieces in order to fit in on the displays of handheld PCs or WAP phones and reduce data transfer. These solutions all contribute to making data and communication accessible while being mobile.

Mobile computer use is, however, more than just a question of communication and information access while being separated from one's desktop, and the usability of mobile devices is not only influenced by technical limitations such as small screens, but also by the relation between their design and their use context. As opposed to desktop computing, users of mobile devices are typically characterized by moving through physical space and interacting with a series of objects and subjects autonomously moving in and out of the user's proximity. This results in use contexts, in which demands for information and functionality changes in accordance to contextual factors such as time, place and social constellations.

The implications of the use context are increasingly being taken into account within mobile HCI. Devices are being equipped with sensors for making them react to how they are being held, touched and moved (see e.g. Hinckley et al. 2000). Spatial, temporal and social *context awareness* are being explored as means of input for customising interfaces to specific situations or locations in time and space (see e.g. Benford et al. 2001) and displays built into eyeglasses are providing mobile users with computerized graphical overlays on the real world, typically referred to as *mobile augmented reality* (see e.g. Feiner et al. 1999). While the properties and use of physical space are often downgraded or left out in interface design for existing mobile technology in favour of more technical issues, the design of context aware and augmented reality interfaces are demanding explicit focus on such contextual factors, as these interfaces are inherently related to the physical space in which they are used.

1.1. Context Awareness and Augmented Reality

Whereas small displays and restricted means of interaction are obvious *limitations* of mobile technology challenging designers to come up with new solutions, the dynamic use contexts of mobile computing does not necessarily constitute a problem, which needs to be solved through design. On the contrary, continuously changing contexts can be viewed as a rich *potential* for interface design.

In making computer systems sensitive to or aware of changes in their use context, interfaces can be designed for adaptation to specific contexts or *habitats* (see Andersen and Nowack, this volume) defined spatially by location in space, temporally by location in time and socially by the presence of other people. Designing interfaces for adaptation to different habitats will not only facilitate reduction or simplification of the information and functionality available in specific situations when moving between habitats, but in doing so also minimize the need for explicitly focusing on and interacting with the device. Viewing context changes as means of input to mobile devices, physical space becomes a part of the human-computer interface, providing the user with information and functionality "just-in-place" (Kjeldskov 2002).

Mobile information services adapted to habitats do not have to run exclusively on mobile phones or pocket-sized PC's but can appear in various forms. In some museums for example, visitors are equipped with infrared headphones providing contextually adapted spoken information related to

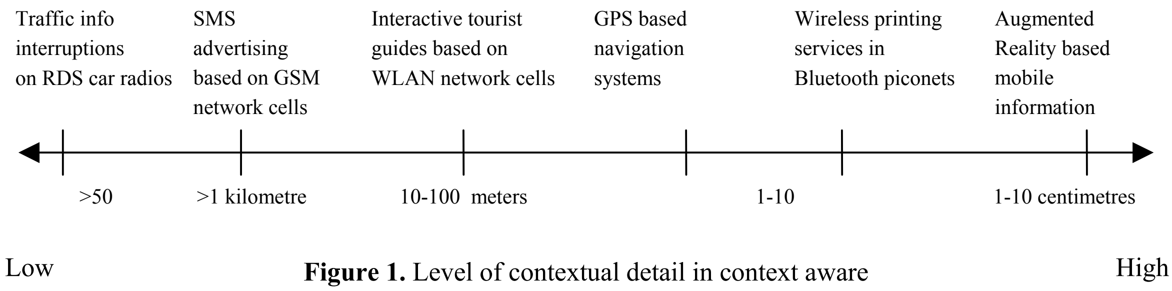


Figure 1. Level of contextual detail in context aware information services for physically defined habitats

the works on display in specific rooms. On RDS enabled car radios, broadcasts are interrupted with spoken traffic messages or supplemented with text-based information on the radio's graphical display, related to a delimited geographical area. Another example is the ability to charge mobile phone users different rates e.g. at home or at work according to their presence within specific GSM network cells.

Using mobile augmented reality, habitats (whether spatially, temporally or socially defined) can be represented in a more sophisticated way through visual augmentation of the user's physical surroundings. Specific locations, objects or persons can be complemented with visual information such as written text, icons, computer graphics, pictures and video, facilitating the reception of and interaction with virtual objects, information and actors, resembling that of their physical counterparts and thus demanding only a limited additional cognitive load. This has great potential in situations where moving ones visual focus away from the physical world should be avoided (such as when driving or walking down the street) or in situations where interaction with computers is naturally related to interaction in physical space (such as in synchronous cooperative work). Other application areas include situations in which information central to interaction in the physical world may be visually hidden or obscured (such as e.g. when manoeuvring a ship through thick fog).

Taking into consideration the role of physical space in computer use, context awareness and augmented reality challenges our usual notions of usability and involves the fundamental traditions of graphical interface design being reconsidered and new paradigms for interaction and interface design being explored.

1.2. Level of Contextual Detail

From the examples above, it is clear that while context aware mobile information services do not have to have augmented reality interfaces, augmented reality interfaces on the contrary depend seriously on a close relation to their physical context. This consequently means that mobile augmented reality cannot be viewed as an *alternative* to context awareness, but is inseparably a sub-ordinated specific variation of this.

Differentiating context aware information services with reference to their relation to physical space, context-aware information services and tools can be plotted onto a continuum of *contextual detail* within spatially defined habitats, describing how level of detail in the interface reflects changes in the user's physical location. At one end of this continuum augmented reality represents a high level of contextual detail while e.g. GPS driven navigation systems represent a lower level of contextual detail. The other end of the continuum, e.g. SMS-based advertising broadcasted when entering specific GSM network cells and broadcasted traffic information interruptions on RDS car radios, represents low levels of contextual detail.

Additional dimensions could be added to this continuum reflecting e.g. the level of *temporal* and *social* contextual detail, refining the picture of relations between existing context-aware information services further.
How accurately physical space and computer interfaces should be related in a specific habitat depends on which role physical space plays in that habitat. If a context aware information service or tool is to provide general information and functionality adapted to a specific room or building, it may be accurate enough to know in which room of a building or which building of a city a user is located, and provide an interface adapted for that room or building.

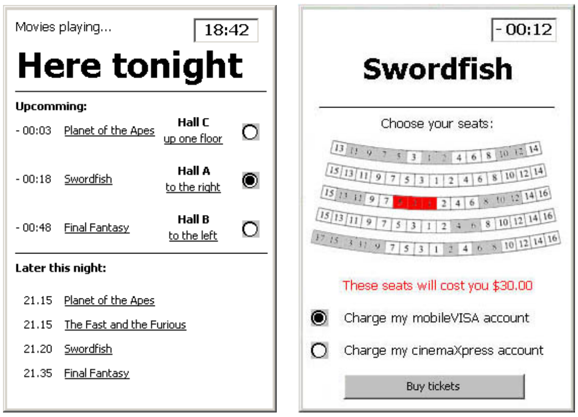


Figure 2. Context aware interfaces for cinema information service available when entering the lobby (left) and waiting outside the hall (right).

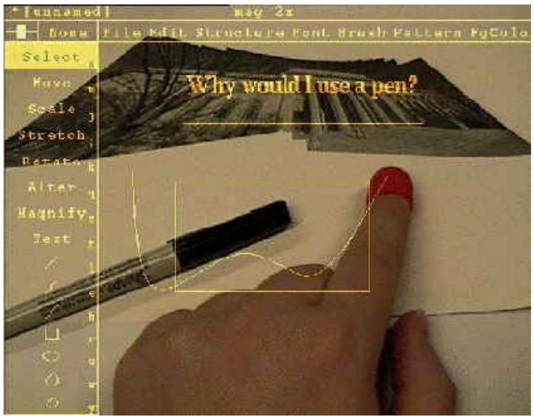


Figure 3. Augmented reality interface for drawing tool enhancing the user's physical desk through computerised graphical overlay (MIT 1999).

In the scenario of designing e.g. a context aware mobile device interface for a cinema facilitating movie information and ticket acquisition, this level of contextual detail could be achieved by making movie information and ticket acquisition available on small-screen devices in the physical space of the lobby and specific halls of the cinema (see figure 2) using e.g. wireless network cells or infrared beaming. Creating a meaningful relation between the interface and physical space at this level of accuracy would not demand an augmented reality interface. However, if a service or tool is to provide highly detailed computer-based information and functionality adapted to specific and delimited areas and objects in a room or building, a higher level of contextual detail is needed concerning the user's whereabouts in physical space and the placement of the interface within this space. In designing e.g. a context aware drawing tool, this level of accuracy could be achieved by spatially tracking the user, superimposing the computer interface onto the physical desktop and facilitating interaction through the use of spatially organized physical tools and objects such as paper, pens, scissors etc. (see figure 3).

Compared to context aware mobile information services for low-level contextual detail, creating a meaningful relation between computer interface and physical space at the higher level of contextual detail exercised in augmented reality, increases the demands for knowledge about the user's contextual surroundings on two levels.

On a *technical* level, the user's location and orientation in space, proximity to objects and subjects surrounding him has to be registered by the computer system with a very high level of precision. While several technologies for motion tracking are already available for virtual reality, appropriate technologies have yet to be developed for mobile augmented reality involving additional demands for facilities such as wireless operation, mobility and range.

On a *cultural and social* level, the properties and use of physical space has to be identified, analysed and described in order to capture qualities and limitations, which should be exploited or supported in the specific interface being designed. In relation to this analysis, it is important to notice that physical space does not have to be replicated in every detail in order to make a useful augmented reality interface. On the contrary, many significant properties of physical space such as e.g. gravity and distance are neither necessary nor appropriate in the majority of augmented reality interfaces in order to make them useful.
Apart from the analysis of the relation between physical space and mobile computing, converting this knowledge into specific interface design constitutes a major issue for the designers of such systems.

2. Analysing the Use of Physical Space for Interface Design

When designing context aware or augmented reality interfaces, designers are faced with two general problems of HCI research.
Firstly, whereas guidelines for traditional WIMP (windows, icons, menus, pointers) based interfaces are widespread (see e.g. Preece et al. 1994, Dix et al. 1998 and Shneiderman 1998), the literature on human-computer interaction does not provide much input on how to design interfaces closely merged with their physical surroundings. Neither is much experience with the usability of such interfaces reported. General guidelines for graphical user interfaces, concerning concepts such as e.g. mental models, mapping, feedback etc. are of course also valid and valuable in relation to context awareness and augmented reality. However, most concepts and notions of HCI research only superficially address the role of physical space and the relation between interfaces and their physical surroundings. Secondly, HCI research may be criticized for primarily supporting *retrospective* measures and explanations of the quality (or lack of quality) in *existing* systems, and not providing a fruitful foundation for the development of *new* good interfaces (Nardi 1996).
Designing human-computer interfaces that differ fundamentally from the WIMP paradigm, such as context aware and augmented reality interfaces challenge both the scope and applicability of existing HCI research and thus call for new approaches to design to be explored. In order to support innovative design, such approaches must provide:

> *"(…) (1) a broad background of comparative understandings over many domains, (2) high-level analyses useful for evaluating the impact of major design decisions, and (3) information that suggests actual design rather than simply general design guidelines or metrics for evaluation. To be able to provide such expertise, we must develop an appropriate analytical abstraction that "discards irrelevant details while isolating and emphasizing those properties of artefacts and situations that are most significant for design"."*
>
> (Nardi 1996:69-70)

One of the problems with this perspective on HCI is the potential conflict between converting theoretical and analytical insight into suggestions for actual design without ending up in the pitfalls of either *dictating* specific solutions, thus limiting creativity and innovation, or leaving it all on creativity, with the risk of losing control of the process and focus. To balance these, an analytical abstraction will be appropriate only if it captures the essence of the design problem in a form both *general* enough for innovation and at the same time *concrete* and *structured* enough for directing the creative efforts.

Designing for context awareness or augmented reality, such analytical abstraction must specifically be able to capture and differentiate the essential properties and uses of physical space in mobile computing in a way that inspires interface design merging the two, without simply just *replicating* the characteristics of physical space.

Inspiration for such analytical abstraction of design can be found in the works of architect and design philosopher Christopher Alexander. In his early work *Notes on the Synthesis of Form* (1964), Alexander presents a perspective on design of general value not limited to architecture and design of physical artefacts but also applicable within other areas such as computer interface and system design. In his principal work *A Pattern Language* (1977), this perspective is elaborated further into a method for describing and categorising design on a level of abstraction that captures the essence of existing form-context relations relevant for new design, while also supporting systematic structuring, interpretation and creativity.

2.1. Form and Context

According to Christopher Alexander, good design is a matter of creating the right *fit* between form and context – form being the solution, the part of the world we can control and manipulate, context being the part of the world defining the problem and making demands on form. Design is thus not a question of form (or context) alone but a question of addressing the creation of *ensembles* fitting the two together. Ensembles of form and context exist on multiple levels. In a game of chess, some moves (the form) fit the stage of the game (the context) better than in other stages of the game. In city planning, improving the city constitutes an ensemble of e.g. the form of expansion fitted to the context of the way the city works, the physical limitations and potentials and the necessity for such expansion (Alexander 1964:16). From this perspective, the world cannot be definitively divided into categories of form or context as both form and context themselves can be considered ensembles of form and context. What is form in relation to a given context may be viewed as context of other forms and visa versa. Subsequently, fit (or misfit) can be obtained either by modifying the form to the context or by modifying the context (including existing forms) to the form. The latter, however, with the danger of breaking down existing fit in other ensembles. According to Alexander, design is typically situated in between these two extremes involving a mutual adaptation of form and context. Because form in itself also influences its own as well as other contexts, fit will consequently always be a dynamic rather than a static property of form-context ensembles, making design a potentially infinite task.

The dynamic relation between form and context necessitates designers to introduce a division between form and context in a given design case or ensemble and specify what defines the problem and what constitutes the domain for possible solutions. Such division will, of course, always be superficial in the sense that it could have been drawn elsewhere. But another division also implies another relation between form and context and thus demands another form. In relation to this, Alexander (1964:17) emphasizes the tendency among designers to stretch the form-context boundary and redesign large proportions of the context rather than simply focusing on the objects, which they are supposed to improve. While this on one hand may be extremely impractical, it is on the other hand stressed as a fruitful way for designers to keep their eyes open to alternative solutions, other than the immediately obvious.

Pushed to the extremes, the development of new form perfectly fitted to its context demands either a complete description of the context or that a number of forms are tried out until a good one is found. Neither approach is realistic in itself. If the context could be described completely there would be no issue of design, only a question of construction. Randomised non-reflected experiments with different forms, on the other hand, cannot be characterised as a process of *design.* According to Alexander, understanding context and experimenting with form are instead inseparable elements in design as insight into the context influences the creation of form as well as experimenting with form influencing the understanding of its context.

In relation to augmented reality user-interfaces for mobile devices, an Alexandrian perspective on design helps understanding of the ensemble of device and use better. First of all it is clear that shifting the use context of computing towards being mobile, requires corresponding form changes in order to maintain fit. Design that may have fitted very well into the context of desktop computing may be less appropriate in the context of being mobile. Supporting text input on mobile phones by means of miniature instead of full size keyboards exemplifies this problem. Furthermore, the use of augmented reality displays implies a displacement of the form-context border, as display and means of interaction usually belonging to the context of interface design now becomes a part of the form domain. Apart from displacing the borderline between form and context, designing augmented reality interfaces for mobile devices also extends the scope of the context domain to involve the use of physical space and other objects and persons in the user's physical surroundings, not necessarily relevant for design previously.

Extending the context may involve additional demands on the form but at the same time also widen the horizon of possible solutions.

Through the process of design, it is of course important to be able to evaluate and elaborate on form. According to Alexander, when people are exposed to misfit between form and context in poorly designed artefacts, they are typically capable of both identifying these and expressing how they are perceived defective. Fit, on the other hand, is neither easy to identify nor to describe because forms fitting their context appear transparent. This has the immediate consequence that fit cannot be described independent of misfit but must be defined as *a form without misfit,* making it problematic to characterise the desired positive properties of a new form without having to refer to the elimination of known misfit. Misfit, in this way becomes a driving force in design through motivation and focus, subsequently introducing a demand for knowledge about the fit/misfit of similar form-context relations (either through practical experience or theoretical insight) among designers.

When designing something fundamentally new such as e.g. augmented reality interfaces, however, examples of misfit can be hard to come up with due to lack of precedence. In this situation, experiments with form play a significant role, not only as a way of creating fit but also as a way of generating misfit, challenging the designer and providing forces pointing towards solutions and extending his insight into the context.

Concrete cases of fit and misfit combined with insight into the context, describing when and why different forms fail or perform well can be systematised as *patterns* of design (Alexander et al. 1977), having the potential of directing designers towards creation of new solutions, recreating qualities of existing forms without replicating the form as a whole.

2.2. Design Patterns

The concept of design patterns deal with some of the challenges of creative design discussed earlier in relation to interface design: how to capture the essence of existing form-context relations (fit and misfit) while supporting creativity and facilitating a systematic approach to the problem domain.

Following Christopher Alexander's (1977) design methodology, a problem domain can be divided into a hierarchy of interrelated "patterns", collectively constituting a "pattern language" for designers across competences and disciplines. Every pattern within this hierarchy potentially relates to the implementation of one or more general patterns and typically points towards a number of more specific, sub-ordinated patterns supporting their own implementation (if such patterns exist). Each pattern represents an empirically based formalised description of central properties and dynamics in the relation between form and context of a specific ensemble. Patterns identify and accentuate known examples of fit and misfit, and guide the designer towards the creation of new form through general examples useful for *inspiration* rather than *duplication.*
In his principal work, *A Pattern Language* (1977), Christopher Alexander provides a hierarchy of 253 interrelated patterns concerning architecture on three levels of abstraction: towns, buildings and constructions.

Though potentially related to a larger theoretical framework, design patterns are not in themselves abstract principles hard to apply in practical design. At the same time, patterns are not so specific, that they lose value in development of *new* forms. On the contrary, design patterns are general concrete guidelines for the creation of fit, unifying insight into form and context pointing towards new form (as illustrated in figure 4) but without dictating specific design.

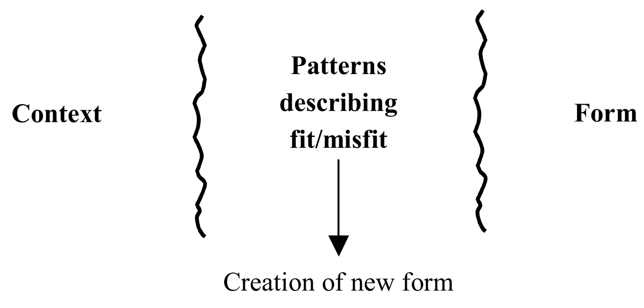


Figure 4. Design patterns merging knowledge about context and form towards new design

Similar to grounded theory, design patterns emerge from empirical observations. But contrary to grounded theory, design patterns do not make up theory as such but are merely structured *descriptions* of form-context relations that have been observed in the field. The process of identifying design patterns in a problem domain, however, very much resembles the process of generating theory from empirical data grounded in the real world. Identifying design patterns thus encourages designers to observe a number of representative and reliable ensembles of form and context in a problem domain, and generate abstractions from this data, which capture the essence and explain the phenomenon on a level of general validity. Like generating theory from empirical data, identifying patterns of design is a long process, requiring designers/researchers to take a step backwards and elaborate on their understanding and ideas or even return to the field for further observations and more focused studies. The methodology of Christopher Alexander (1977) does not provide much operational input on how to manage this process. However, the structure and interrelation of Alexandrian design patterns provide a valuable template aiding the formulation of patterns from ones empirical observations.

Alexandrian design patterns follow a uniform structure and graphical format, which makes them highly consistent (illustrated in figure 5). Each pattern is headed with a brief name and a photograph exemplifying its application. Following the photograph, the pattern is related to possible parent patterns, which it may contribute to the realisation of. This is followed by a concentrated problem statement, describing the essence of the problem addressed by the pattern in one or two short sentences (formatted in bold). Subsequently, the empirical background of the pattern and detailed insight into the context along with examples of fit and misfit are presented. This part is often accompanied by further pictures, drawings, diagrams etc. substantiating the empirical foundation. Finally, the solution to the problem (the essence of the pattern) is described in a few brief sentences (also formatted in bold) often supplemented with a general graphical representation. Each pattern is terminated with references to possible sub-ordinated patterns. A single design pattern typically takes up 3-5 pages.
The uniform format of Alexandrian design patterns has a number of advantages in the design process. First of all, the graphical layout and consistent use of photographs and text formatting makes collections of patterns (or pattern languages) easy to browse and be inspired by without having to read them all word by word. Secondly, the structured format facilitates design patterns being read on several levels of abstraction according to the need for input at a given time.

For browsing and overview, patterns can be read primarily through their names and photographs, pinpointing design issues and capturing possible applications of the pattern. For this use, the photographs heading each pattern play a significant role, often expressing visual concepts more effectively than words could and stimulating the user's memory of a pattern. Selecting good photographs illustrating one's design patterns is thus critical and should be regarded as an important component of the pattern.
For brief and summarised insight into the design issue addressed by a pattern, designers need only read the short statements describing the problem and pointing out solutions. For this use, it is important that these sections are both brief and clear, and do not delve too deeply into the background of the pattern.
Finally, for a full description, covering the empirical and theoretical foundations of a pattern, design patterns can, of course, be read in their full length. For this level of abstraction, it is important that the pattern provides insight, which substantiates both the problem statement and the suggested solution.

In practical use, designers will most likely shift back and forward between these levels of abstraction during the design process and e.g. 1) browse headings/photographs for a quick overview, then 2) read the problem and solution statements of those patterns, which seem immediately interesting, then 3) read selected patterns in their entirety for additional insight and finally 4) implement selected patterns using primarily their problem statements and suggested solutions while browsing between them using their headings and photographs.

2.3. Design Patterns as User Interface Requirement Specifications

Whereas the concept of design patterns has received much attention within software engineering and object oriented programming, less focus has been brought to this approach within human-computer interaction. Though several references to the work of Alexander exists within mainstream HCI literature (see e.g. Norman and Draper 1986, Norman 1988, Winograd 1996), only a few examples exists of HCI researchers actually applying the concept of design patterns and pattern languages in interface and interaction design. This, even though the concept of design patterns as originally intended within architecture has much more obvious relations to user interface design than to software engineering in respect to the focus on users and design meeting the demands and potentials of a use context. Among the most central work on design patterns for HCI are the investigations of Tidwell (1999) and Borchers (2001), illustrating the potentials of design patterns as user interface requirement specifications.

Through a comprehensive collection of interrelated patterns, Tidwell (1999) shows how the overall concept of a highly interrelated web of patterns can be applied to interactive software, describing highly specific issues of interaction design with a broad range of applications such as “going back to a safe place” and “disabling irrelevant things” while leaving out issues about implementation and user interface technologies. Exemplifying the strength of a pattern approach for capturing “timeless” examples of fit and misfit in interaction design, the majority of these patterns successfully balance the requirements of being neither too specific nor too general for application outside the context within which they were identified.

Whereas Tidwell does not strictly follow the format of Alexandrian patterns described earlier, making her patterns less inspiring to browse and rather superficial regarding their empirical background compared to the patterns of Alexander, Borchers (2001) carries the format of Alexandrian design patterns all the way through. In his patterns on the design of interactive exhibitions, dealing with more general interface issues than Tidwell such as “domain appropriate devices” and “immersive display”, Borchers shows how patterns for HCI can benefit from following both the structure and the level of abstraction characterising Alexander’s design patterns for architecture, facilitating a high level of interpretation.

While both the works of Tidwell and Borchers, however, focus primarily on the *identification* of patterns on interaction and interface design from existing HCI design, the actual *applicability* and *usefulness* of such patterns in the creation of *new* interface design still needs to be investigated and reported.

In the following sections, the practical applicability of design patterns for human-computer interaction is explored through a specific design case, dealing with the design of an experimental augmented reality prototype for distributed cooperative work. A number of empirically identified design patterns for computer supported cooperative work are presented, followed by a description exemplifying the actual application of these patterns in an augmented reality interface.

Compared to the HCI patterns of Tidwell and Borchers, the patterns described in this chapter are more closely related to the use of physical space than to the experience with specific design solutions of existing well-performing human-computer interfaces. The reason for this is twofold. First of all, the use of physical space plays a significant role in the type of interface being designed. Secondly, the developed prototype explores interface design within an area with limited precedents. The design is thus not aimed at recreating specific qualities of existing tools as much as it aimed at recreating the qualities related to the use of physical space and solving some general misfit in existing tools.

3. Design Case: Augmenting a Habitat for Cooperative Work

Workspaces for cooperation such as offices, meeting rooms, workshops etc. can be viewed as spatially, temporally and socially defined habitats, facilitating specific cooperative activities at different times according to the people present or not present. Such workspaces often involve people in a workgroup moving around and artefacts being closely related to specific physical locations at specific times. Furthermore, physical space is frequently being reconfigured for a variety of use purposes: working individually or in groups, conducting presentations, holding meetings, organising and storing shared resources, socialising etc. When a workgroup is physically distributed, many of these properties and potentials of physical space are lost.

A variety of computer-based tools for supporting cooperative work (CSCW) exist for supporting the creation of distributed shared workspaces facilitating sharing of files and applications as well as video-based interpersonal communication on ordinary desktop computers. While overcoming some of the problems of distributing a workgroup physically, these tools typically fail to support users being mobile in, and take advantage of the properties of, physical space, usually characterising cooperative work. Instead of providing the user with a transparent virtual space, within which cooperation can exist, CSCW tools thus typically constitute new tools to be operated, in order to facilitate cooperation. This consequently results in an additional layer being introduced between the users and objects of a workgroup, creating demands for an intuitive representation of the system's conceptual model, restricting interaction and introducing a rather solid boundary between objects in physical and virtual space respectively.

Using mobile augmented reality interfaces to augment habitats for cooperative work through the creation of distributed virtual workspaces merged into physical space, introduces new potentials for the design of CSCW tools. Potentially relieving some of the problems of existing CSCW tools, mobile augmented reality interfaces for supporting distributed cooperation could enable a stronger relation between information systems and physical space as well as a higher level of mobility in the physical world while also being present in a distributed virtual 3D space.

Exploring this hypothesis, an experimental augmented reality prototype for distributed cooperative work was designed and implemented. The interface design of this prototype was based on requirement specifications in the form of design patterns, describing the use of physical space in cooperative work in relation to examples of fit and misfit in existing CSCW tools.

3.1. Identifying Design Patterns for Computer-Supported Cooperative Work

During 1999, 12 design patterns were identified and used as interface requirement specifications for an augmented reality interface supporting a physically distributed workgroup. The basis of analysis leading to the identified design patterns was three-fold. 1) Literature studies on the role of physical space and artefacts in our daily lives generally (Hall 1966, Tuan 1977, Tillich 1933) and in cooperative work specifically (Kaptelinin 1996, Kuutti 1996, Obata and Sasaki 1998) provided a theoretical background for the empirical observations and analysis of collected data. 2) Empirical observations of cooperative work activities in physical space at the university of Aalborg (meetings, presentations, workshops etc.) with and without the use of computer-based tools provided specific insight into the use of physical space and artefacts. 3) Usability evaluations of a broad range of tools for computer-supported cooperative work (Microsoft NetMeeting, Teamwave Workplace, SGI InPerson, VirtualU, FirstClass, etc.) pointed out examples of fit and misfit in existing tools.

From the empirical observations, a number of distinctive characteristics and properties of cooperation in physical space were extracted and organised in a simple list of *issues*. Each issue was then discussed and reconsidered in relation to the others resulting in some being merged and others being left out. The remaining issues were then organized in a hierarchy and related to the theoretical perspective on physical space and labelled with a number of theoretical concepts. Similarly, the identified usability problems as well as examples of successful design in existing computer-based tools for supporting cooperation was listed, grouped and sub-ordinated to each other. Where possible, usability issues were related to the corresponding use of physical space and labelled with theoretical concepts from the literature. Subsequently the two lists were merged into a hierarchy of preliminary outlines of 12 interrelated design patterns associated with the physical distribution of a workgroup, formulated in terms of physical space use with references to specific design. Finally the outlined design patterns were refined further, reformulated and supplied with an extended body discussing the problem addressed by the pattern and pointing out avenues for possible solutions. Photographs capturing the focus of each the pattern were then added.

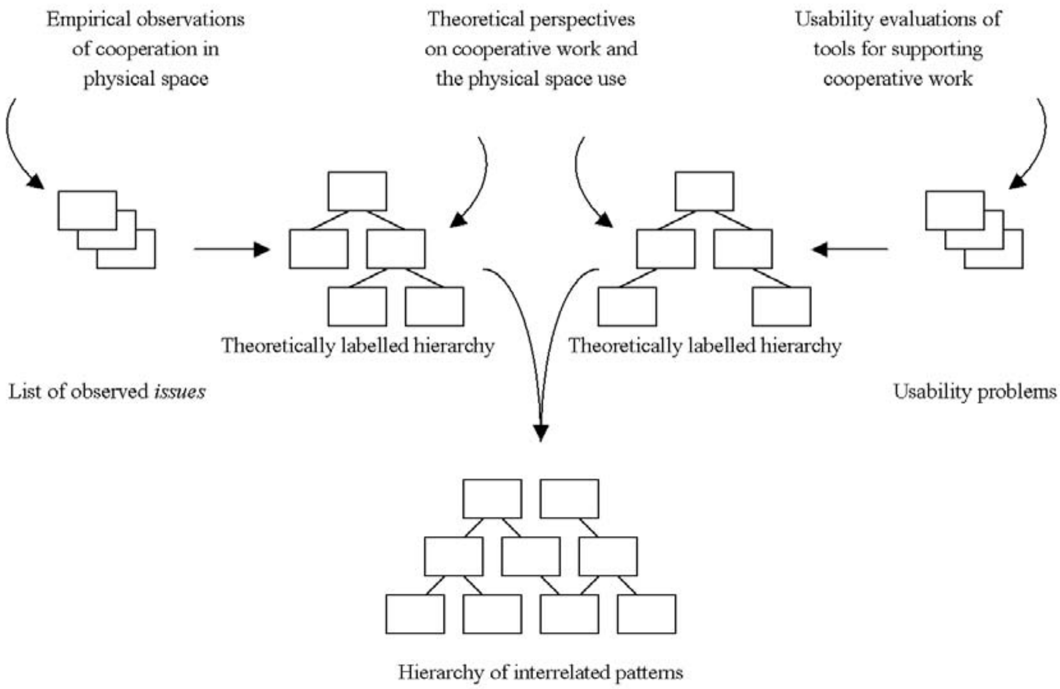


Figure 5. The process of identifying patterns of cooperative work

4. Outlines of a Pattern Language for CSCW

From the original 12 design patterns identified, the following six had most significant impact on the interface design of the implemented prototype, and will be presented in this chapter in summarised form.

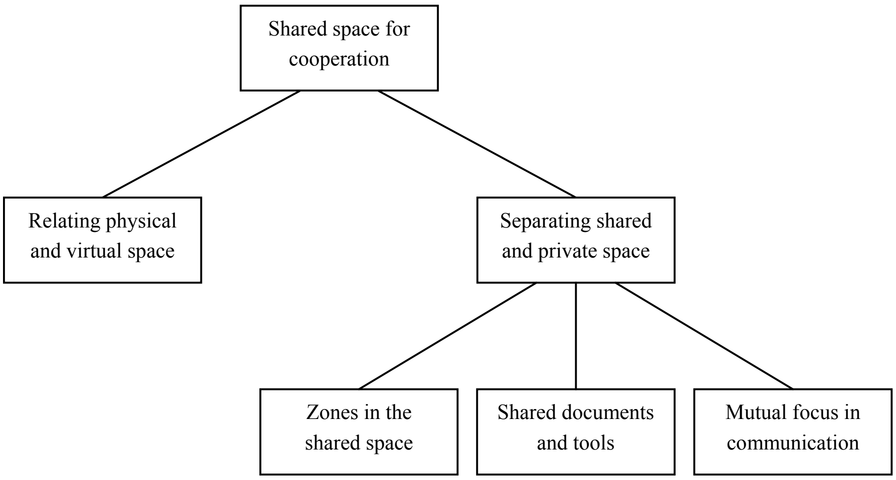
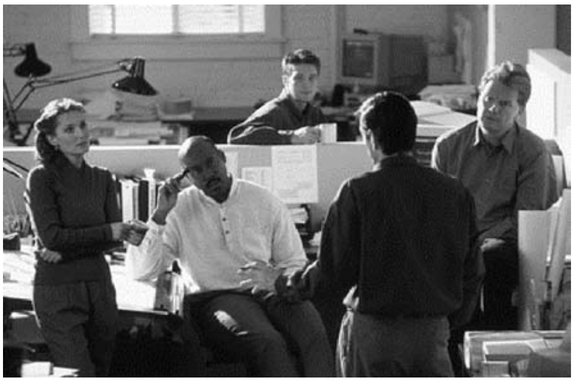


Figure 6. Hierarchal structure of the identified design patterns for CSCW

Within the context of augmented reality-based interfaces for cooperative work these patterns can all be subordinated to the realisation of Borchers' (2001) pattern H8 on Augmented Reality:

> "In situations where your system aims to recreate a real experience, use as much real instead of virtual environment as possible. Instead of trying to recreate all aspects of reality inside an artificial environment, augment the real environment with interactive technology that is not usually available otherwise"
>
> (Borchers 2001: 127-128)

1. Shared Space for Cooperation



When a group of people are working together, it is important to have access to a shared space for cooperation facilitating members of a workgroup relating to each other, sharing documents and engaging in communication, presentations, meetings, etc. When a workgroup is distributed physically, this space for cooperation is lost.

Observing cooperative work, it is obvious that the joint activity of a workgroup can be coordinated. Coordination typically takes place in three different ways: *scripted* through means such as plans, calendars and timetables, *communicatively* through written or spoken language or *instrumentally* through knowledge of each others actions. These ways of coordination require that members of a workgroup have access to shared resources concerning their joint activities, can communicate and interact with each other and are aware about the actions and activities of the workgroup as a whole. The closer a workgroup cooperates the more it depends on means for coordinating. Under normal circumstances, physical space plays a vital role as a highly versatile medium for such mechanisms. When a workgroup is distributed physically, this medium is lost. Observing the use of tools for distributed cooperative work indicates that while scripted and asynchronous communicative coordination is typically supported very well through access to e.g. shared folders, e-mail based communication and discussion forums, it is hard to support instrumental and synchronous communicative coordination. While chat relays, video communication and application sharing have potentials in this direction, current implementations of these technologies are still rather limited.

Therefore:

Create a computer-based shared space for cooperation by connecting the individual workspaces of people in a distributed workgroup to each other. Within this space, facilitate information access and flexible communication and interaction among the group. If possible, support the maintenance of peripheral awareness about the presence and activities of others.

Create such shared space for cooperation by *relating physical and virtual space (2)* and *separating shared and private space (3)* with reference to *zones in the shared space (4), shared documents and tools (5)* and *mutual focus in interpersonal communication (6).*

2. Relating Physical and Virtual Space

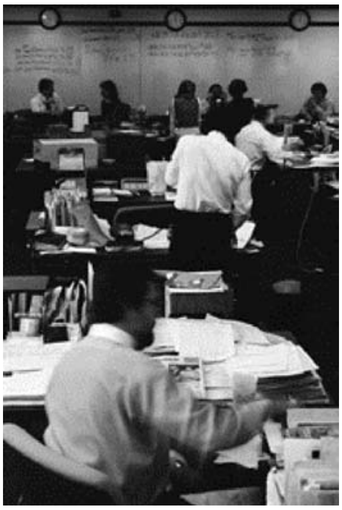
The use of computer-based tools or media in distributed cooperative work introduces a hard boundary between virtual objects and physical space, restraining the relation between the two. Each member of a workgroup consequently has to maintain an informational (virtual) space and a physical space, as well as an arbitrary relation between the two.

While physical space facilitates objects being spatially organised and related to each other in a meaningful way, according to their location in space, this facility is restricted when dealing with a mixture of physical and virtual objects.
Observing the use of tools for computer supported cooperative work indicates that physical and virtual spaces are highly separated, and that user's spend considerable efforts on maintaining a relation between the two, displacing the focus of activity away from the cooperation itself.
The lack of relation between physical and virtual space often results in a messy, confusing and unnatural organisation of the elements in the interface. In tools for supporting a distributed workgroup, shared objects, representations of other members of the workgroup, control panels, etc. are typically crowding the user's desktop in a very unstructured fashion without any relation to the physical space surrounding the tool. Such badly organised interfaces limit the usability of the shared space for cooperation, because fundamental operations such as changing focus from one object to another (whether virtual or physical), interacting with shared tools or communicating with remote users cannot be done without consciously focusing on interaction with the *system.*
The problem can be described as lack of context awareness, making interaction in the physical world surrounding the system "invisible" to the system and visa versa. How accurate interaction in physical space should be reflected in virtual space depends on the specific context of use. Some applications such as augmented reality may require a very close relation while others may require only a superficial relation in order to support interaction.

Therefore:

Strengthen the relation between physical space and virtual objects by making the computer system context aware on a level of detail matching the specific need of the cooperative work activities being designed for. Design the means of interaction in a way that makes interaction in physical space a part of the human-computer interface.

3. Separating Shared and Private Space



In a shared space for cooperation, it is important that members of a workgroup are not forced to be together at all times, but can separate from each other and retreat to a space of their own. The separation of shared and private spaces is critical. Lack of separation breaks down the private space while too much separation isolates people from the workgroup.

According to Tillich (1933), the presence of private space constitutes a significant foundation for people's ability to act in shared space. Without private space, people have nowhere to retreat to and feel secure but are constantly public, resulting in a feeling of root- and restlessness. Goffman (1959) addresses the same issue through the concepts of front- and backstage, while Tuan (1977) describes the problem as a matter of facilitating the creation of a *place* as opposed to *space:* a secure base, in which people are their own masters, in charge of the spatial layout and organisation of space and artefacts.
Observing the use of shared physical workspaces confirms the fine balance between shared and private space and furthermore indicates, that the separation is often dynamic, exemplified by e.g. people keeping the doors to their office open or closed towards the hallway at specific times.
In cooperative work, the importance of separating shared and private spaces is not only related to the individual need for a place of ones own. It is also simply a matter of facilitating individual work activities, relatively undisturbed by the presence and activities of others.
The separation of shared and private space is traditionally not an issue taking up much focus within CSCW, as the problem may remain invisible until private space is threatened by e.g. the creation of distributed shared spaces, exposing it to other people.

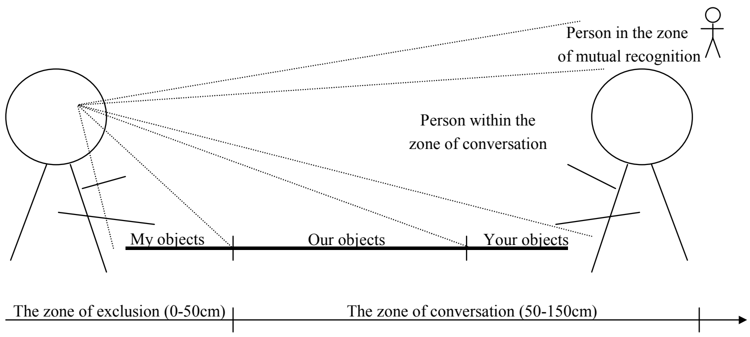
Therefore:

Provide each member of a workgroup with a private space facilitating individual work, clearly delimitated from the shared space. Make the private workspace half-open towards the shared workspace so that peripheral contact to the other members of workgroup can be maintained without people being able to keep a private workspace under surveillance.

4. Zones in the Shared Space

In cooperative work, it is important to be able to organize the shared space of a workgroup in a simple and comprehensible way, facilitating differentiation of objects belonging to the individual members and objects being shared by the group as well as easy exchange of private and shared objects among the members of the group.

In physical meetings around a central table, space is often divided into a number of zones, having a number of properties for organizing the shared space. Objects located within a person's intimate distance (15-45 cm) (Hall 1966) are considered belonging to this person while objects located within a person's personal distance (45-120 cm) but not within other people's intimate distance are considered shared by the people surrounding these objects. Objects within the social distance of a person (1,2 – 3,65 m) are also considered shared by the group while objects or persons within a person's official distance (3,65 m –) may belong to other groups. The division of shared space into zones supports an implicit reduction of complexity, clearly distinguishing between private and public objects as well as persons within close or remote proximity. The figure below illustrates such division of space in a physical meeting into specifically named zones (Obata and Sasaki 1998).



Therefore:

Divide the shared workspace into a number of zones corresponding to the zones of physical meetings facilitating intuitive identification of my objects, our objects, your objects as well as close and remote persons. Make the objects belonging each member of a group visible to other members within close proximity and facilitate easy exchange of objects between persons within close range.

5. Shared Documents and Tools

Cooperative work typically involves the use and organisation of a large number of shared documents, diagrams, archives, drawings, etc. and the joint interaction with tools for modifying these. When a workgroup is physically distributed, the access to shared documents is limited and the cooperative use of tools is made difficult.

Cooperating in a shared physical space, shared documents etc. constitute a common frame of reference and a central means of asynchronous communication within the workgroup both through their informational content and their spatial relations, grouping, location, etc.
When a workgroup is distributed physically, the members of such group usually end up with multiple outdated versions of shared documents, and the common spatial relation between shared documents is lost, breaking the common frame of reference. In tools for supporting a distributed workgroup, it is thus important to support not only access to shared documents but also facilitate the members of a group jointly organising these documents in an appropriate way.

In relation to shared documents, tools for modifying these documents, whether individually or in groups, also plays a significant role in cooperative work as "mediating artefacts" (Kuutti 1996), supporting the transformation of the objects of cooperation and the instrumental coordination of activity. Observing the joint use of shared tools in physical as well as virtual space reveals a significant problem related to the regulation of turn taking: who is in control of the shared tool at a given time and how is the control taken by or passed on to another? In physical space, turn taking is usually regulated by cultural rules and cues, facilitating quick and easy handover. In tools for supporting distributed cooperative work, turn taking on the other hand often involves a structured procedure to be carried out, facilitating control but limiting flexible use.

Therefore:

Within the shared workspace, make it possible to store, organise and retrieve shared information, and support the use of appropriate tools for modifying these individually or in groups. Make it possible to organise shared documents in a flexible way similar to the way one would organise physical documents: spatially, in stacks etc. If a shared tool can only be operated by one at a time, facilitate an easy regulation of turn taking.

6. Mutual Focus in Interpersonal Communication

In physically distributed meetings, it can be difficult to maintain the mutual focus in interpersonal communication, which is implicitly established in physical space by virtue of visual orientation and body gestures.

Observing communication and interaction in physical meetings, the importance of visual orientation and body gestures of others becomes very clear. Establishing an implicit mutual focus and facilitating instrumental coordination of joint interaction among the members of a group, are a means to support the basic control of communication and joint interaction such as e.g. turn taking (passing the word on to another speaker) and indexicality (looking or pointing at someone or something while talking or interacting).
When people are physically distributed, the mutual focus characterising interpersonal communication in physical space is impeded, making it difficult to maintain a common frame of reference among members of a workgroup. This not only restricts interpersonal communication through a distributed shared space due to lack of properties such as mutual gaze-direction, but also lowers the usability of shared tools and documents in synchronous cooperation.
Reducing the means of instrumental coordination, people consequently have to compensate for the lack of mutual focus and body gestures through other modes of operation such as meta-communication like “I now do this while looking at that…” or “do you see where I am pointing now…?” etc. This need for meta-communication further strains the channel of verbal communication and moves the focus of activity away from the objects of cooperation towards conscious operation of the tool itself.

Therefore:

Design the shared space for cooperation so that the gaze-direction and gestures of the distributed users such as pointing is reflected and represented in a natural and meaningful way. If possible, introduce into this representation some kind of reciprocity of gaze-direction, resembling the feeling of face-to-face contact in physical space.

5. Applying Patterns to Augmented Reality Interface Design

In the following section, it is described how the six patterns of design presented above were applied to actual interface and interaction design. "Being There" is an experimental prototype, exploring the use of mobile augmented reality interfaces in relating a spatial habitat to a virtual information space on a high level of contextual detail. The prototype exemplifies how the visual augmentation of a private office for distributed cooperative work could be designed for access through a mobile device interface, exploiting the potentials and use of physical space in cooperative work. The illustrations show the user's perspective of the interface in an office already rich with physical objects and artefacts. The prototype was developed in Macromedia Director 7.0 with supporting software for TCP/IP communication and serial data input from input devices. The prototype setup consisted of two Sony Glasstron PLM-S700 semi-transparent head-mounted displays, two Polhemus IsoTrakII, a 5th Glove Dataglove and a 3D mouse.

5.1. Creating a Shared Space for Cooperation

The pattern on *creating a shared space for cooperation (1)* motivates the overall creation of tools for supporting distributed cooperative work. In the implemented prototype, such shared space for cooperation was created by augmenting the physical offices of each member in a workgroup with a computerised graphical overlay of a virtual 3D world, running on a central server and shared over high-speed network. This design first of all facilitated a much larger graphical interface than on traditional desktop monitors, leaving room for much more information without cluttering the interface. Wearing lightweight, semitransparent head mounted displays users at the same time remained mobile in the physical space of their offices while simultaneously also being present in the virtual shared space of the workgroup

The specific design of this shared space was realized through *relating physical and virtual space (2)* and *separating shared and private space (3)*.

5.2. Relating Physical and Virtual Space

Supporting the relation between virtual objects and physical space, the implemented prototype was made context aware on a high level of spatially contextual detail, generating an impression of computer-based documents and tools being present alongside with their physical counterparts. This is illustrated in figure 7, showing the user's desk, populated with physical and virtual objects.

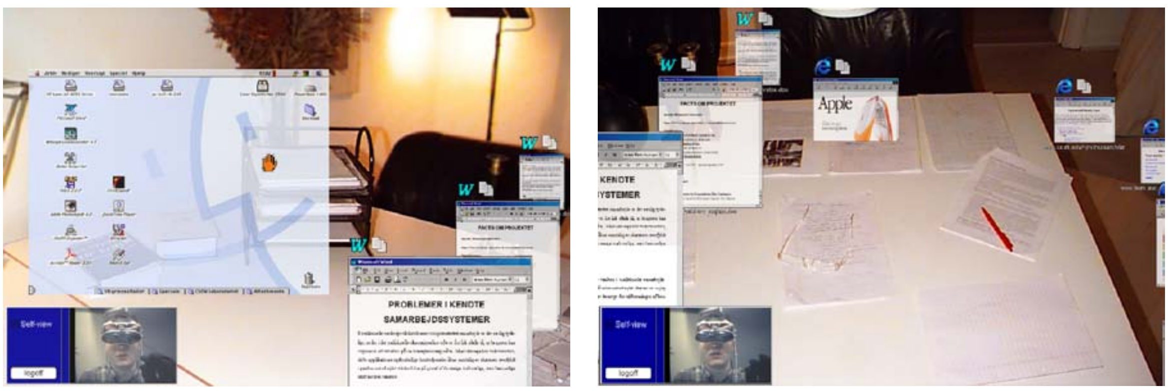


Figure 7. Physical desk populated with physical as well as virtual objects

By turning his head, the user can change the view of the virtual world in the same way he would orientate in the physical world, preserving a consistent spatial relation between the two. In the lower left corner is a self-view, showing the video feed being transmitted to the other users of the system.

Blurring the boundary between physical and virtual space further, the interface design facilitates virtual objects being spatially organised in all three dimensions, being grouped, stacked, pushed further away or dragged into closer proximity. Additionally, virtual objects could have been directly related to physical objects, creating an even stronger relation between physical and virtual space. This was, however, not implemented.
Interaction is done by means of a dataglove facilitating virtual objects being grabbed, moved and modified with ones hand, in more or less the same way one would interact with physical objects. Rectifying some of the limitations of interaction in physical space, the user is, however, equipped with "stretchable" virtual arms, making him capable of reaching objects in virtual space over a longer distance than in physical space.

5.3. Separating Shared and Private Space

Implementing the pattern of *separating shared and private space (3),* the virtual space merged into the users physical office is divided horizontally into two overall areas. The area of the virtual space surrounding a user's physical desk is considered private and is not visible to the other users in the system, while the area facing away from the desk is shared among the group. This is illustrated in figure 8, showing a view of the private space on the left and the boundary between private and shared space on the right. Physically this boundary is marked with the end of the desk. In the computerised overlay this location, is marked/enhanced with a vertical line.

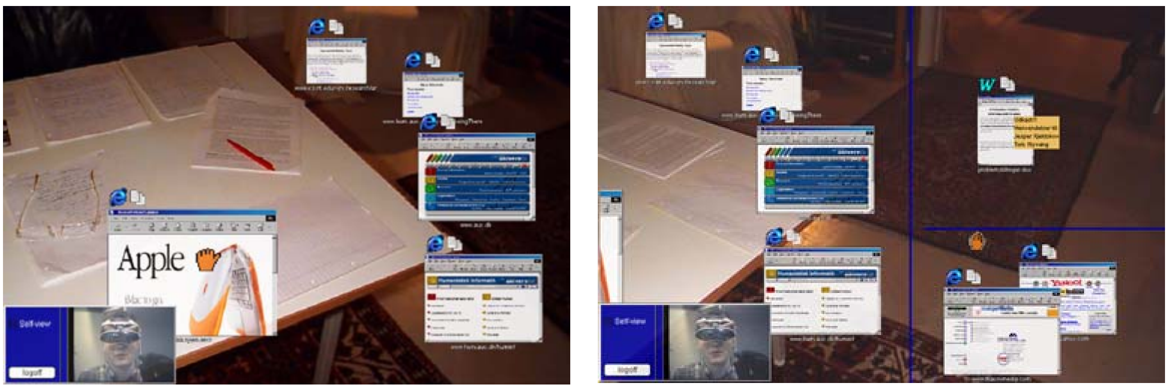


Figure 8. The boundary between private (left) and shared space (far right)

The implemented layout of the interface to a large degree resembles the typical division of physical space in both private and shared offices, supporting peripheral contact to the workgroup while working in ones private space and facilitating an easy transition between private and shared spaces by simply turning around on ones chair. Exploiting this design further, documents can easily be exchanged (copied) among the group by simply dragging them from private to shared space and visa versa.

The specific design of the shared space is realised through *dividing the shared space into zones (4),* and supporting *shared tools and resources (3)* and *mutual focus in interpersonal communication (6).*

5.4. Dividing the Shared Space Into Zones

The layout of shared space (facing away from the physical desktop) is organised vertically in accordance to the pattern describing *zones in the shared space (4).* The implemented design more or less directly resembles the zones characterising a physical meeting.
Looking down at his lap, the user has visual access to documents within the shared space, corresponding to the documents located right in front of him in a physical meeting (figure 8, left). This resembles the zone of exclusion, containing documents belonging to a specific person but visible to others in proximity. Looking upwards, the user has visual access to the shared documents of the workgroup, corresponding to the documents located in the zone of conversation between people in a physical meeting (figure 9). These documents can be organised and manipulated simultaneously by all users. Further upwards in the interface, the user sees the documents belonging to other members of the group: your documents (figure 9, right). The representations of the zone of exclusion of remote users are located immediately below their respective live video image (see section 5.6), supporting a natural mapping between the two.

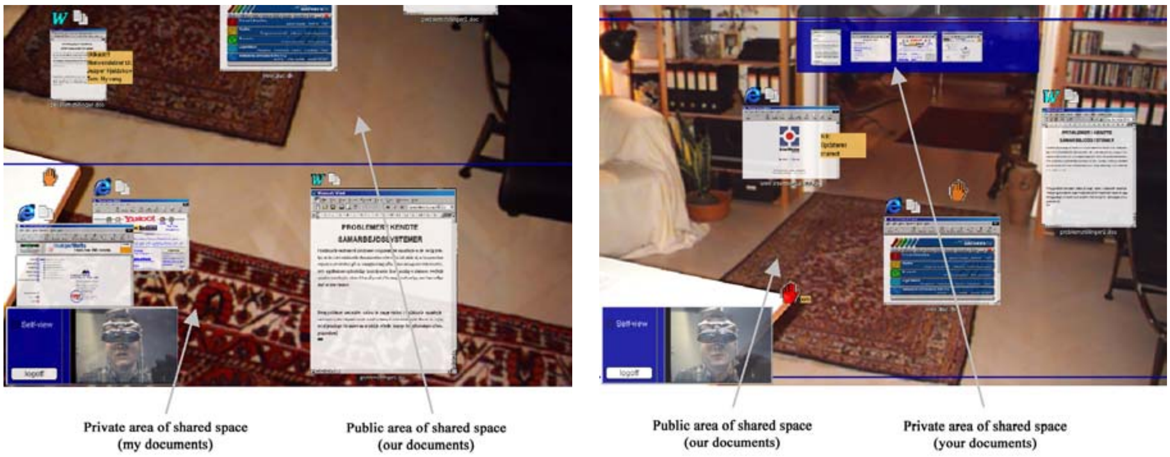


Figure 9. The division of shared space into zones

Though designed in accordance with the zones of proximity in mind, the design of the shared space does not aim to resemble physical space in all aspects. Contrary to physical space, all users see the shared space from the same visual perspective, preventing users having to compensate for the mutual displacement characterising a "real" 3D world directly facilitating the use of spatial indexicality in communication such as referring to "the document down to the right".

5.5. Supporting Shared Documents and Tools

In accordance to the pattern on *shared documents and tools (5),* the shared space for cooperation merged into the user's physical office supports the joint spatial organisation and interaction with computer based documents and tools.

Reducing the traditional boundary between documents and tools in computer artefacts, the interface design of the prototype facilitates documents (e.g. web pages or word documents) being editable in their corresponding tools by simply dragging them into closer proximity (figure 10).

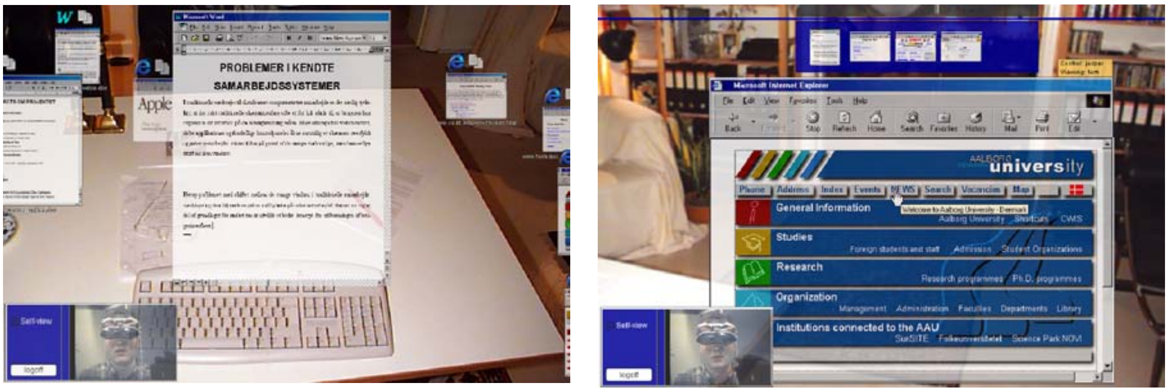


Figure 10. Private and shared tools embedded into physical space

While tools in the private space (figure 10, left) are not visible to the group, tools in the shared space (figure 10, right) are by default shared among the users. As the multi user tools available in the prototype (word processors and web browsers) can only be operated by one user at a time, regulations of turn taking have been implemented. Resembling the turn taking in physical space, control over a shared tool is taken by another user simply by starting to interact with it. While this regulation of turn taking may result in anarchy and a constant battle for control, it on the other hand facilitates turn taking being controlled by cultural rules rather than through formalized procedures.

5.6. Supporting Mutual Focus and Interpersonal Communication

Looking straight into the shared space, the user has visual access to the other members of the workgroup facing the shared space through live video feeds (figure 11). In accordance to the pattern on *zones in the shared space (4),* the vertical location of these video images resembles the location of other people's faces in a physical meeting.

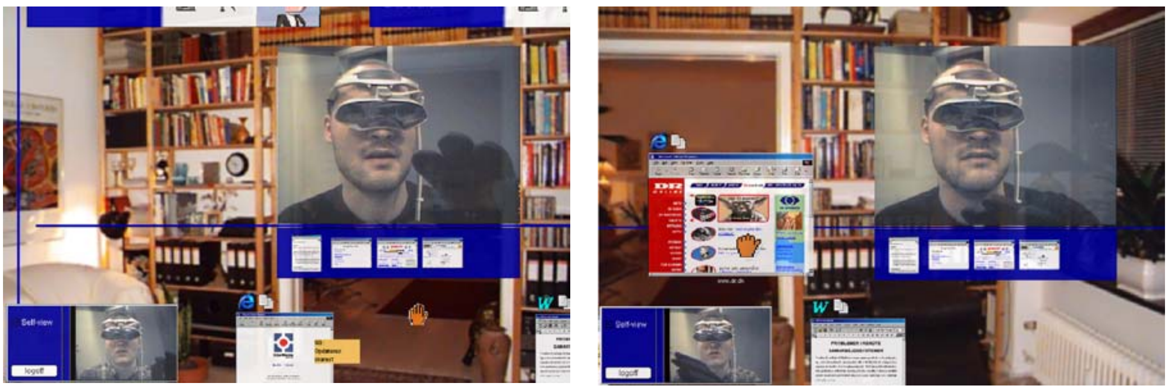


Figure 11. Gaze-direction represented by the horizontal location of live video in shared space

Implementing the pattern on *mutual focus in interpersonal communication (6),* a remote user's gaze-direction in the shared space is reflected by the horizontal location of his video image in the shared space, thus causing video images of remote users to slide from left to right according to their orientation. This is illustrated in figure 11.

While very different from the way physical space works, this design has a number of advantages. First of all, it supports a simple cue for revealing the visual orientation of other users, making it possible to determine what remote users are looking at in the shared space. Secondly, an approximation of face-to-face contact is supported by the reciprocity of video image locations, meaning that if I look towards the video image of you, you will also be faced with the video image of me, providing valuable assistance in the establishment of interpersonal communication. This reciprocity is not available in traditional video based communication tools. Thirdly, if a user is not facing the shared space, no video image will be available, preserving the privacy of private space. In accordance with the pattern on *zones in the shared space (4),* remote users facing away from the shared space are represented as distant avatars in the zone of mutual recognition (in the top of the interface), supporting peripheral awareness (figure 12).

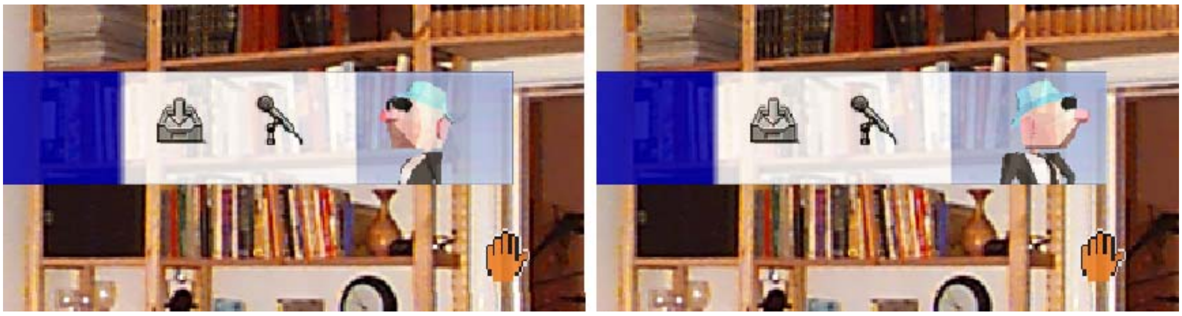


Figure 12. The gaze-direction of remote users represented through the orientation of avatars

Providing each user with a distinguished pointer in shared space supports the representation of pointing in shared space but does not represent actual body gestures.

6. Lessons From Being There

The presented prototype exemplifies what a mobile augmented reality interface for distributed cooperative work *could* look like in order to support the properties and use of physical space, described earlier. The lessons learned from developing Being There can be divided into three sections:

1. Augmented reality interface design
2. Mobile augmented reality
3. Using patterns in interface design

6.1. Augmented Reality Interface Design

Apart from being tested extensively by the developers, the design was evaluated through interactive exhibitions at Intermedia Aalborg in August/September 1999, during which a total of 25-30 people unfamiliar with such interfaces tried out the system for approximately 10 minutes each. During use, people were asked to “think aloud”, communicating their comprehension and intentions when interacting with the system. Following use, people were informally interviewed about their experience with the system.

Figure 13. Informal usability evaluation of Being There

The evaluation showed that even though the graphical performance of the system was not optimal, the design demanded little instruction being carried out. Users typically had no problems understanding the relation between physical and virtual space and using the system for spatially organising virtual objects in physical space as well as orientating in virtual space by moving physically. They often walked right up and used the system. Some users, however, expressed a wish for linking virtual and physical objects, others expressed wishes for real 3D representation and stereoscopic projection, which was not facilitated in the prototype.

Users showed little or no problems differentiating between private and shared space and exchanging documents with each other. Some, however, expressed a wish for a clearer graphical identification of the boundary between the two, which could have been affected by the lack of natural demarcations of private and public areas in the physical space of the exhibition area. The division of shared space into zones was also easy to comprehend. Users expressed that they found the location of video images straight ahead of them natural and easy to use, and that the location of documents virtually in their lap supported a feeling of ownership.

While users easily engaged in interpersonal communication through the video links, the representation of gaze-direction was, however, reported as difficult to understand and rather confusing by first-time users. Suggestions of other designs were made, e.g. indicating viewing direction through a small compass below each video image. Users communicating through the system for a longer period of time, however, found the representation of gaze-direction very practical and easy to use, expressing that it actually supported their communication and interaction.

Though very intuitive at first, direct manipulation of virtual objects using the dataglove was reported demanding, as it required large physical arm movements for even simple operations. While maybe related to the relatively low frame rate and high response time of the prototype, similar problems are also reported in the use of high-end virtual reality installations without these limitations (Kjeldskov 2001), indicating that absolute input devices may simply be problematic for some types of interaction in virtual space. In both cases, using a relative pointing device would demand smaller physical movements.

The evaluation of Being There indicates that augmented reality interfaces exploiting the potentials and use of physical space and populating it with virtual objects have a high level of immediate usability, although fundamentally different from traditional human-computer interfaces. The virtual implementation of qualities found in physical space does, however, not demand physical space to be replicated in all aspects, but can be achieved through design inspired by it leaving designers room for creativity, exemplified by the unnatural representation of gaze-direction. On the contrary, implementing natural properties of physical space may result in limited usability as exemplified by the glove-based interaction technique.

6.2. Mobile Augmented Reality

Though not suitable for mobility outside a dedicated physical office due to the use of stationary electromagnetic tracking devices, the implemented prototype provides input on the design and usability of eye-glass based mobile augmented reality, relating physical space and user human-computer interfaces on a high level of contextual detail.

Addressing the challenges of mobile HCI described in the beginning of this chapter, Being There shows how separating the computing device from the graphical interface and embedding the display into the user's eyeglasses can increase the limited screen real estate of mobile computers. Covering a large percentage of the user's visual field of view at all times does, however, not necessarily mean that it should be completely filled with graphical information at all times. In most mobile use contexts e.g. when walking down the street, the primary advantage of the relatively large screen real estate of an augmented reality interface is not the amount of information potentially squeezed into the interface. It is rather the ability to capture the user's visual attention and locate small bits of information at the appropriate spatial location within his visual view, enabling the user to receive information without having to pull out and hold a physical device. As physical space is typically already abundantly populated with objects, artefacts and people (illustrated by figures 7–12), designers of mobile augmented reality interfaces must balance the weight of embedding virtual objects into physical space without creating an explosion of complexity.

Exploring the potentials of context awareness in mobile device interfaces, Being There shows how virtual information can be merged successfully into specific locations surrounding the user, preserving a dual focus on the computer interface and the physical world. While the prototype primarily exemplifies how a *spatially* defined habitat could be represented through a context aware mobile information service, mobile augmented reality could also facilitate the representation of temporally and socially defined habitats through similar design, taking these contextual factors into consideration.

In relation to the limited means of interaction characterizing mobile devices, the use of augmented reality in Being There does not indicate an evident path for solutions. On the contrary one could claim, that the problem of interacting with mobile device interfaces is in fact increased in the light of augmented reality by removing the usual relation between physical input device and visual output device. While walking down the street, how does one interact with virtual objects floating in space? Apart from the use of simple pointing devices, interesting areas of research count the use of speech (Sawhney and Schmandt 2000), eye-movements (Duchowski and Vertegaal 2000) and even brainwaves (Hjelm and Browall 2000) as means of input, potentially applicable within the context of mobile augmented reality.

6.3. Using Patterns in Interface Design

The employment of Alexandrian design patterns in the development of Being There exemplifies how the properties and use of physical space can be analysed and described in a form supporting creative interface design for augmented reality. In the specific design case, the methodology of Alexander constituted an appropriate analytical abstraction by describing interface requirement specifications through a series of design patterns, balancing the need for facilitating creativity in new design while approaching the design problem in a systematically and structured way. The identified patterns specifically played a threefold role as *design tool, communication tool* and *structure tool.*

As a *design tool,* the patterns reduced the complexity of the specific form-context ensemble in focus and pointed towards new solutions by identifying and describing key issues of the context and existing forms relevant for design while discarding irrelevant details. As no specific solutions were dictated, the patterns challenged the designers to be innovative within the frame of the patterns.

As a *communication tool,* the patterns supported the accumulation of the problem domain analysis and mediated this knowledge on to the process of design. During design, the patterns worked as a common frame of reference in communication within the design team, supporting a high degree of subjective interpretations within a common frame of understanding.

As a *structure tool,* the web of patterns divided the form-context ensemble into a series of interrelated tangible design problems to be addressed explicitly. Each isolated pattern furthermore supported the structure of the design process by dealing with a clearly delimited area of the form-context ensemble.

While design patterns are great for supporting communication and a structured approach to creativity, they are in themselves, however, too general for implementation without further efforts being made. At some point of the design process, patterns must thus inevitably be complemented with additional and more detailed requirement specifications, specifying exactly *how* the interface should look and work and *how* it should be constructed in order to do so. In this perspective, design patterns may be most valuable in the *early* phases of a software development process.

6.4. Limitations of This Work

As the primary focus of this project was the preliminary studies of the design potentials and problems of mobile augmented reality, the prototype was never intended to make up a fully implemented system suitable for real-world use, but should rather serve as a *proof of concept.* Full-scale usability studies of the interface design in real distributed cooperative work setting were thus not conducted, naturally limiting our ability to assess whether or not the design really has the intended qualities of physical space and would perform well over a longer period of time.

7. Conclusions

In this chapter, I have presented mobile augmented reality as a possible human-computer interface approach for context aware information services with a very detailed relation between physical and virtual space. In the light of a number of limitations within HCI research in relation to the creation of *new* interface design, I have presented the concepts and methodology of Christopher Alexander, suggesting the relations between ensembles of form and context being described as patterns of design. Using this approach, I have shown how a mobile augmented reality interface for supporting distributed cooperative work can be designed from empirically identified design patterns, capturing the role of physical space in cooperative work.

The implemented prototype shows that physical and virtual space *can* be related in a mobile augmented reality interface and that locating virtual objects in the user's physical surroundings *can* be useful in a mobile device interface. The evaluation of the prototype indicates that such interfaces *can* be comprehended and utilized by ordinary users.

Acknowledgements

The author thanks Tom Nyvang who participated in the identification of the original 12 design patterns and the development of the prototype. Acknowledgements are also paid to Ellen Christiansen and Erik Granum for engagement and enthusiastic supervision on the project and their participation in constructive discussions. Finally, I thank Jeni Paay for comments on a late draft.

References

Alexander C (1964) *Notes on the Synthesis of Form.* London, Harvard University Press.

Alexander C, Ishikawa S, Silverstein M (1977) *A Pattern Language*. New York, Oxford University Press.

Benford S et al. (2001) Citywide: supporting interactive digital experiences across physical space, Proceedings of the Mobile HCI 2001 Workshop on Human Computer Interaction with Mobile Devices, IHM-HCI 2001, Lille, France.

Bergman E (ed.) (2000) *Information Appliances and Beyond.* London, Morgan Kaufmann.

Borchers J (2001) *A Pattern Approach to Interaction Design.* New York, John Wiley & Sons, Ltd.

Dix A, Finlay JE, Abowd GD, Beale R (1998) *Human-Computer Interaction.* Second Edition, London, Prentice Hall.

Duchowski A, Vertegaal R (2000) Eye-based Interaction in Graphical Systems: Theory and Practice. In Course Notes no. 5, Siggraph 2000, New Orleans.

Feiner S, MacIntyre B, Höllerer T (1999) Wearing It Out: First Steps Toward Mobile Augmented Reality Systems. In Onto, Y. and Tamura, H. (eds.) (1999) *Mixed reality – Merging Real and Virtual Worlds.* Berlin, Springer-Verlag.

Goffman E (1959) *Presentation of Self in Everyday Life.* Garden City, N.Y. Anchor.

Hall E (1966) *The Hidden Dimension.* New York, Doubleday.

Hinckley K, Pierce J, Sinclair M, Horvitz E (2000) Sensing Techniques for Mobile Interaction, in Symposium on User Interface Software and Technology, CHI Letters 2(2):91-100.

Hjelm S and Browall C (2000) Brainball – using brain activity for cool competition. Demo at NordiCHI 2000, Stockholm, Sweden.

Kaptelinin V (1996) Computer-Mediated Activity: Functional Organs in Social and Developmental Contexts. In Nardi B (ed.) (1996) Pp 45-68.

Kjeldskov J (2001) Combining Interaction Techniques and Display Types for Virtual Reality. In Proceedings of OzCHI 2001. Edith Cowan University Press, Churchlands, Australia, Pp 77-83.

Kjeldskov J (2002) "Just-in-Place" Information for Mobile Device Interfaces. In Paternò F (ed.) (2002) *Human Computer Interaction with Mobile Devices.* In Proceedings of 4th International Symposium, MobileHCI 2002, Pisa, Italy, September 2002. Berlin, Springer-Verlag.

Kuutti K (1996) Activity Theory as a Potential Framework for Human-Computer Interaction Research. In Nardi B (ed.) (1996) Pp 17-44.

Mackay WE and Fayard A (1998) Designing Interactive Paper: Lessons from Three Augmented Reality Projects. In Behringer et al. (eds.) (1998) *Augmented Reality: Placing Artificial Objects in Real Scenes: Proceedings of IWAR'98.* A K Peters, Pp 81-90.

MIT MediaLab (1999) The Wearable Computing Project. http://www.media.mit.edu/wearables/lizzy/augmented-reality.html

Nardi B (ed.) (1996) *Context and Consciousness: Activity Theory and Human-Computer Interaction.* Cambridge, MA, The MIT Press.

Norman DA, Draper SW (eds.) (1986) *User Centered System Design.* London, Lawrence Erlbaum Associates, Inc., Publishers.

Norman DA (1988) *The design of everyday things.* New York, Doubleday.

Obata A and Sasaki K (1998) OfficeWalker a Virtual Visiting System Based on Proxemics. In Proceedings of CSCW 98, Seattle, Washington, USA, November 14-18, 1998. New York, ACM, Pp 1-10.

Preece J, Rogers Y, Sharp H, Benyon D (1994) *Human-Computer Interaction.* Workingham, Addison Wesley.

Sawhney N and Schmandt C (2000) Nomadic Radio: Speech and Audio Interaction for Contextual Messaging in Nomadic Environments. In ACM Transactions on CHI, 7(3):353-383.

Shneiderman B (1998) *Designing the User Interface: Strategies for Effective Human-Computer Interaction.* 3rd Edition, Reading, MA, Addison Wesley.

Tidwell J (1999) A Pattern Language for HCI. http://www.mit.edu/~jtidwell/interaction_patterns.html

Tillich P (1933) Dwelling, Space and Time. In *On Art and Architecture.* New York, Crossroad, Pp 81-85.

Tuan Y (1977) *Space and Place – The Perspective of experience.* Minneapolis, The University of Minnesota Press.

Winograd T (ed.) (1996) *Bringing Design to Software.* New York, ACM Press.

Published in Halskov K. (ed.) (2003) *Production Methods: Behind the Scenes of Virtual Inhabited 3D Worlds*. Berlin, Springer-Verlag, pp 18-38

Interaction Styles in Development Tools for Virtual Reality Applications

Jesper Kjeldskov and Jan Stage
Department of Computer Science
Aalborg University
Denmark
{jesper,jans}@cs.auc.dk

Jesper Kjeldskov has a background in the humanities with a master's degree in humanistic computer science. He is currently a Ph.D. student working with human-computer interaction for virtual reality and mobile computers.

Jan Stage is associate professor in computer science. He holds a Ph.D. degree in computer science from University of Oslo. His research interests are in usability testing, HCI design, and software engineering.

Virtual reality display systems reflect an emerging technology that is used to visualize virtual three-dimensional (3D) worlds. This article describes and evaluates tools for developing software applications that are targeted at virtual reality display systems. The evaluation focuses on the relevance of command language or direct manipulation with graphical representation as the fundamental interaction style in such a tool. The foundation of the evaluation is an empirical study of two development processes where two tools representing each of these interaction styles was used to develop the same virtual reality application. The development tool that employed a command language was very flexible and facilitated an even distribution of effort and progress over time, but debugging and identification of errors was very difficult. The tool that employed direct manipulation enabled faster implementation of a first prototype but did not facilitate a shorter implementation process as a whole due to e.g. lack of support for writing code for repetitive operations. The limited advantage of using direct manipulation for developing virtual reality applications is further explored through comparison with a successful direct manipulation tool for developing interactive multimedia applications targeted at traditional desktop computers. From this comparison, a number of central issues for user interface design of highly graphical development tools are pointed out for future design.

Methods and guidelines for user interface design embody certain computer technologies. This also applies to interaction styles. Interaction based on a command language was a relevant solution with the character-based display. Direct manipulation emerged from the potentials of the graphical workstation and personal computer. This inherent relation between interface design and computer technology implies that our established guidelines and experiences are challenged when new technologies emerge.

Virtual Reality display systems are used to create and visualize virtual three-dimensional (3D) worlds. This technology is emerging, and practically relevant applications are being developed for a broad range of domains. As the use of the technology is increasing, there is an increasing demand for tools for developing applications.

The purpose of this article is to compare and discuss the relevance of classical interaction styles for tools that are used to develop virtual reality applications. The focus is on the process of developing actual virtual reality applications. More specifically, we compare the potentials of a development tool based on a command language with one that is based on direct manipulation. The discussion is structured in the following way. First we review selected literature on command language and direct manipulation as classical interaction styles. We then describe the virtual reality display technology in order to specify the target platform we are aiming at. The comparison of the two classical interaction styles is based on an empirical experiment. Hereafter we describe the design of that experiment, and emphasize the relevant results. The empirical findings are then related to the authors' experiences with a successful development tool for traditional 2D multimedia applications. Finally, we conclude the discussion and point out avenues for further work.

1. Interaction Styles for Development Tools

The interaction style is a key determinant of the design of the user interface. Many discussions on the advantage or disadvantage of a certain design relate to this characteristic. The options available for design of this characteristic have been denoted as: command language, menu selection, form filling, and direct manipulation (Shneiderman 1998). Below, we will refer to these as the classical interaction styles.

The development of a virtual reality application includes an essential task where we construct the 3D world that is visualized by the application. The fundamental interaction style of existing tools that support this task employs either direct manipulation or a command language in the form of a textual programming tool. Menu selection and form filling are also employed but only for secondary interactions that deal with limited issues, e.g. the specification of properties of a certain object that is manipulated directly on an overall level. Based on these priorities, the discussion below deals only with command language and direct manipulation.

The literature on human-computer interaction includes numerous contributions to the question whether direct manipulation is superior to command languages. Much of this is description of advantages whereas the amount of empirical evidence is very limited (Benbasat and Todd 1993). An early contribution compared file manipulation commands in MS-DOS with Macintosh direct manipulation. This study concluded that the Macintosh users could perform the manipulations faster, with fewer errors, and they were more satisfied with the interface (Margono and Shneiderman 1987). A similar study where command line and direct manipulation was compared concluded that the users of direct manipulation made only half as many errors and were more satisfied. In this study, the time to perform the tasks turned out to be comparable (Morgan et al. 1991).

The limited number of reports from empirical studies of command language and direct manipulation seem to indicate an advantage in terms of error rate and user satisfaction. When it comes to time to complete a task, the conclusions are more varied. Our intention in this paper is to examine these expectations in relation to tools for developing virtual reality applications.

2. Virtual Reality Applications

A virtual reality application that visualizes a 3D world consists of a number of mathematically defined 3D models that are covered with colors or textures, e.g. pictures or video images. The 3D models are spatially distributed in a three-dimensional coordinate system that the user can experience as a 3D world by viewing the 3D models from a given point in the coordinate system. The correct perspective is rendered real-time by a graphics computer and projected by means of a display system as illustrated in figure 1. Navigation in the virtual world is accomplished by means of position tracking or a specialized interaction device. Tracking the position of user's head also ensures that the correct visual perspective is calculated. Interaction with objects in the virtual world is typically supported by techniques for selecting and modifying 3D objects by simply "grabbing" them just like one would do in the real world. The 3D experience requires shutter glasses worn by the user allowing separate images to be projected to the user's left and right eye and thereby creating an illusion of 3D.

Figure 1. A virtual 3D world

A virtual reality application may use a multitude of display systems to visualize the virtual 3D world. Examples of display systems are traditional desktop monitors, head-mounted displays, holobenches, large wall-mounted displays or caves with various numbers of sides. These display types represent the array of technologies for creating immersive experiences that range from "looking at" a virtual 3D world to "being in" that virtual world (Shneiderman 1998). The six-sided cave (Cave Automatic Virtual Environment) is currently the display system that offers the greatest level of immersion into a virtual 3D world. The user is placed in a small cubic room, measuring approx. 3 meters on all sides, in which the computer-generated images are back-projected on all four walls, the floor and the ceiling, cf. figure 2.

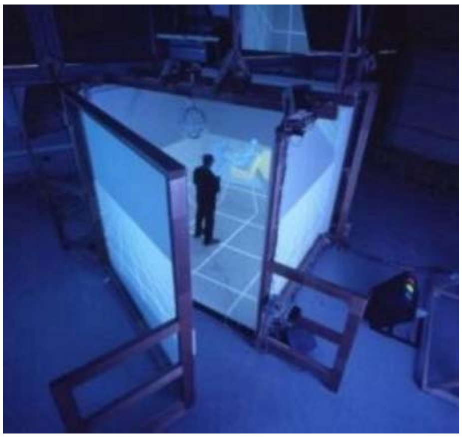


Figure 2. Outside the six-sided cave

Figure 3. Inside the six-sided cave

The benefits of the six-sided cave for exploration of virtual 3D worlds lay in the vividness of the virtual environment projected and the very high degree of immersion. This is caused by the freedom of movement that is possible inside the cave and the large horizontal and vertical field of view covered with images. Exploration of the virtual world is much more natural in a six-sided cave compared to any other display system because the user can move around physically and look in any direction without breaking the illusion of being in a computer-generated world. The primary downside is that physical objects and the user's body itself may occlude the images, thus locally breaking the visual illusion (cf. Kjeldskov 2001).

Virtual reality applications displayed in a cave are very different from many traditional computer applications. First, the user interface is completely surrounding the user and is presented in 3D as opposed to conventional 2D interfaces covering only a fraction of the user's physical surroundings. Second, the types of applications running in a cave are typically offering a complete virtual 3D world for exploration as opposed to traditional tools for office or home use. Third, applications running in a cave are by default both highly graphical and interactive.

2.1. Two Categories of Development Tools

Although virtual reality applications are fundamentally different from typical applications, they are not developed in the cave itself. Virtual 3D worlds are usually developed on ordinary – yet powerful – desktop computers with traditional 2D displays. The existing tools for developing virtual reality applications fall in two different categories.

The first category can be characterized as a classical programming approach, since the creation and manipulation of the virtual world and the objects in it is specified in a command language. Within this approach, special libraries for creating cave applications are available for C and C++. One of the most widely used binary libraries for developing virtual 3D worlds is CaveLib. This library enables development of highly immersive 3D interfaces for projection in a cave, or any other virtual reality display system, as well as implementation of interaction techniques for 3D interaction devices. For preview purposes, CaveLib offers a simple tool for representing the cave display and simulating simple 3D interaction, cf. figure 4.

Using CaveLib to develop an application is not very different from developing any other graphical application in a typical programming language.

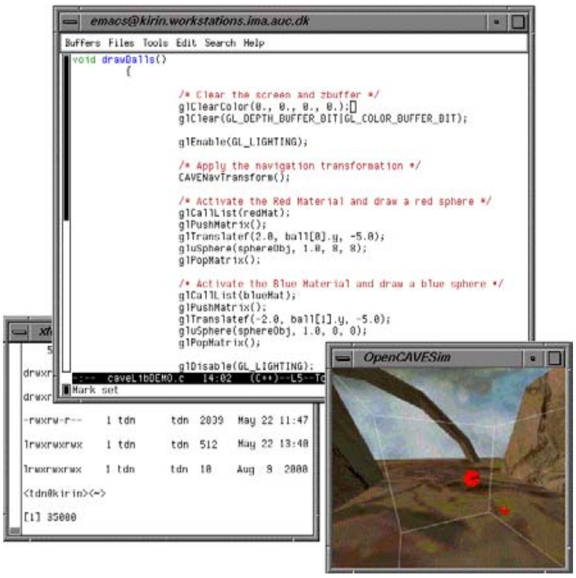


Figure 4. Development with CaveLib

When developing a virtual reality application using a command language tool like CaveLib, the developer constructs a program code pointing at a number of geometry files and specifying the layout of a virtual 3D space as well as the functionality of the application. This code is constructed in a simple text-editor and is typically distributed in a number of script files. To see if the code is working properly, the developer has to compile all the scripts and run the application either in the cave itself or in the preview-tool. If the code contains errors or otherwise needs to be modified, the developer returns to the text-editor.

The second category of tools for developing virtual reality applications can be characterized as a graphical representation and direct manipulation approach. One of the few professional tools in this category is dvMockup.

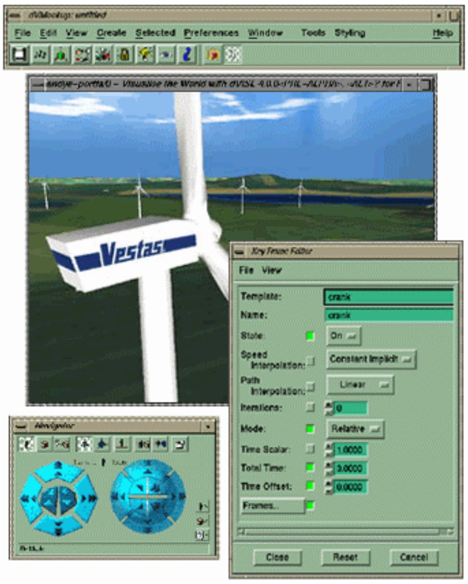


Figure 5. Implementing using dvMockup

This tool enables the developer to create an application by directly manipulating the objects of the virtual 3D world within the preview window along with the use of menu selections and fill-in forms, cf. figure 5. Using dvMockup, implementing an application for the cave is done without doing any actual programming.

When developing a virtual reality application using a direct manipulation tool like dvMockup, the developer imports a number of geometry files and locates them in the virtual 3D space of the application. This is done either by direct manipulation in a workspace-window or by specifying data in forms. The functionality of the application is created and modified by selecting 3D objects and applying behaviors through menu-selection. Through the workspace-window, the developer can continuously see if the application is working properly.

3. Experimental Design

An empirical experiment was conducted to inquire into the research question that was raised in section 1. This section describes the design of that experiment.

Tools: We briefly surveyed potentially relevant tools for implementing virtual reality applications and related this to the fundamental aim of comparing direct manipulation tools with programming tools. Based on the survey and the facilities available, we selected two established tools: dvMockup, a direct manipulation tool that enables people without programming experience to create a virtual reality application, and CaveLib, an advanced programming tool that facilitates development of virtual reality applications characterized by high performance and flexibility. In addition, we selected a new and promising programming tool, VR Juggler, which extends CaveLib with a more flexible and modular structure and open source architecture. The first two tools were already installed, configured, and used extensively by other developers and researchers who could be consulted when technical problems arose. The third tool, VR Juggler, was acquired right before the beginning of the experiment and there were no experiences with it.

Participants: A development team of three persons and the two authors of this article planned and designed the experiment, whereas the development team conducted the implementation phase. The three developers had recently completed a master degree in computer science/computer engineering. Thereby, they had considerable experience and knowledge about programming in general. They had previously taken a one-semester course on computer vision and virtual reality and worked with projects within that subject. They received a one-day introduction to the tools used in the experiment but had no experience with them.

Overall task: The comparison of the three tools was based on solution of the same overall task. The overall task was to develop a virtual reality application that visualized a maze in which a user could move an avatar around by means of an interaction device. This task was specified in detail by dividing it into 14 milestones. Thus, the overall task was solved when all milestones were met. The milestones involved tool and cave set-up (milestone 1 and 2), implementation of a simple application (milestone 3 and 4), implementation of the application visualizing the maze (milestone 5 to 8), interaction techniques to facilitate motion of the avatar (milestone 9 to 12), and adjustment (milestone 13 and 14).

Hypothesis: Based on the literature on interaction styles reviewed in section 2 above, we developed the following hypothesis: The direct manipulation tool is superior to the programming tools in terms of the efforts required to implement the a virtual reality application that is specified by the overall task.

Experimental procedure: When the planning phase of the experiment was complete, the implementation phase started. This phase was planned to last three weeks but was extended with a couple of days because of technical problems. Each member of the development team was assigned one of the three tools and should use it to produce the best possible solution to the overall task. During the implementation phase, they were not supposed to communicate with each other about their work, problems, and solutions.

Data collection: The primary means for data collection were private diaries written by each developer, cf. (Jepsen 1989, Naur 1983). After a day of work on the implementation, each developer used about an hour to describe the work done and its relation to the 14 milestones, the problems faced, and the time spent on tasks related to each of the milestones. A checklist that emphasized the points that should be touched upon supported the daily writing of the diary. One week into the implementation phase, the diary entries produced so far were reviewed enforces the use of the checklist and increase consistency. The three diaries amount to a total of 45 pages (Hougaard et al. 2001).

Data analysis: The primary dependent variables were work practice and development effort. Yet, in this article we focus only on development effort. Based on the diaries, we have calculated and compared the efforts spent on completing the different milestones of the overall task. The results of this are presented in the following section.

Limitations: The design of this experiment imposes certain limitations on our results. Firstly, the members of the development team were not highly experienced in implementing virtual reality applications. Secondly, The overall task defined a specific application that should be implemented. These limitations imply that the results primarily facilitate relative as opposed to absolute conclusions about efforts. Thirdly, the diaries of the three developers were different. In order to handle this they were reviewed after one week of work. The fourth limitation was that two of the tools were established on the development platform whereas the third was not even configured and there was no experience with its use. The developer who worked with this tool ended up spending a considerable amount of effort on issues related to installation and execution. Therefore, we ignore this part of the experiment in our comparison below.

4. Findings

In this section, we present and discuss the findings from the experiment with CaveLib and dvMockup. The developer who used CaveLib was able to meet all milestones, but the navigation technique specified was changed due to usability issues. The developer who used dvMockup was not as successful, since collision detection could not be implemented satisfactory. However, the final solution was acceptable. The development time spent using CaveLib amounts to 42,3 hours, whereas the time spent using dvMockup amounts to 37,8 hours. The total time spent on development with the two tools thus differ only 12%. The distribution of time spent on each milestone does, however, reveal significant differences between the programming and direct manipulation approaches. This distribution is shown in figure 6. Below we will highlight interesting points from this distribution.

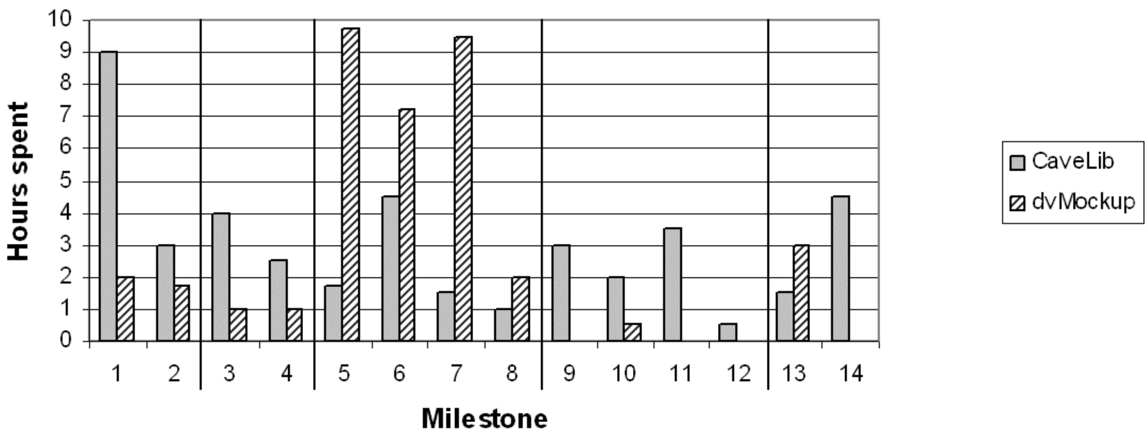


Figure 6. Development time spent using CaveLib and dvMockup.

Setting up the development tools and the cave (milestone 1 and 2) amounted to a total of 12 hours spent on CaveLib whereas only 3,75 hours was spent on this with dvMockup. Thus the developer who used dvMockup only needed about 30% of the time spent using CaveLib. Setting up CaveLib demanded a series of separate tools to be configured for individual tasks, e.g. scripting, compiling and previewing, as well as creation of a number of configuration files on both the workstation used for development and the graphics computer that was executing the display system for the cave. With dvMockup only one tool had to be set up, and when an application was running on the workstation, only a few scripts were needed before it was also operational in the cave.

Implementation of a preliminary application with the purpose of testing the development and target platform and the connection between them (milestone 3 and 4) took 6,5 hours using CaveLib but only 2 hours with dvMockup. Again, for dvMockup this is only about 30% of the time spent using CaveLib. Thus up to milestone 4 it is clear that the direct manipulation approach supports a faster kick-off on the development process.

Implementation of the primary application, which was the maze specified in the overall task (milestone 5 to 8), was done in 10,3 hours using CaveLib. With dvMockup the same milestones required 27,5 hours. So here we see the same pattern where one tool requires only 30% of the time spent with the other tool. Yet this time the roles are reversed, as CaveLib is the favored tool. Thus the programming approach seems to facilitate a more effective process in this part of the implementation. The major reason for the considerable amount of time spent with dvMockup is that the tool provides no direct support in a situation where making and running a simple program might avoid numerous repetitions of simple operations. For example, the developer using dvMockup faced the task of *manually* inserting 800 identical cubic objects into the virtual 3D world, whereas the developer using CaveLib could perform the same task simply by writing a small piece of code. This limitation becomes even more serious when we take the question of scale into consideration. If we compare a small application to a large one, the difference in amount of work will occur precisely on milestone 5 to 8 whereas the remaining milestones will largely be unaffected. Therefore, the difference between the two tools on these milestones will even be more significant if we expand the scale of the application being developed.

Implementation of interaction techniques (milestone 9 to 12) took 7,5 hours with CaveLib while and only 2,5 hours using dvMockup. This is a 30% reduction in favor of dvMockup.

The time spent implementing interaction techniques with dvMockup is, however, influenced by the availability of supporting software. In a related project, a considerable amount of time had recently been spent developing “off-the-shelf support” for implementing interaction in dvMockup in order to facilitate a general reduction of development (Kjeldskov 2001). Had this support not been available, the time spent on these milestones would definitely have increased, but we will not attempt to estimate by how much. In CaveLib, all interaction techniques were implemented from scratch.
However, this had the advantage that the interaction technique specified in the overall task was actually implemented. With dvMockup it was necessary to select one of the available techniques, which did not fulfill the specification completely. If the implementation in dvMockup should have fulfilled the requirements completely, additionally programming on device driver level would have been necessary.

Final adjustments of the applications (milestone 13 and 14) took 6 hours for CaveLib while only 3 hours was spent with dvMockup. The larger amount of adjustments of the CaveLib application primarily consisted of correcting errors with the scaling of 3D objects. This was necessary to make the objects fit properly for projections in the cave. This kind of errors was absent in the application developed with dvMockup.

5. A Successful Direct Manipulation Tool

The findings described above raises a number of questions. What caused the direct manipulation approach to be outperformed by a command language approach? Is it really fair to conclude that direct manipulation is simply not a well-suited interaction style when developing virtual reality applications? Or what exactly were the limitations of dvMockup?
In order to address this question we will take a closer look at a widely used tool for developing interactive multimedia successfully employing direct-manipulation and compare a number of central characteristics of this tool with the corresponding characteristics of dvMockup.

For several years Macromedia Director has been considered state of the art within development tools for interactive multimedia targeted at traditional desktop computers. Much like dvMockup, the interaction style in Director is primarily direct manipulation and fill-in forms but with facilities for scripting/programming. Director is based on a film/theatre-metaphor putting the designer/developer in the "directors chair". The end-user's screen is represented as a rectangular surface (the stage) on which the designer/developer can place and directly manipulate the different graphical elements: graphics, video, sound etc. (the cast) of the application using the mouse and the keyboard. The functionality of the application developed is defined on a central timeline (score) and in a number of accompanying scripts and behaviors linked to he appropriate elements on the screen. Surrounding the stage is a number of tools and fill-in forms for manipulating the elements of the application. Further tools are available through menus or buttons at the top of the screen. The application being developed can rapidly be previewed directly and accurately on the workstation used for development, as display and interaction devices used by the designer/developer are comparable to those of the end-user.

Based on our experiences from teaching Macromedia Director to university students and industrial software developers over the last 4 years, we have observed that the direct manipulation interaction style employed Director performs very well and fit with the developers needs during the different phases of the system development process.
But how come that direct manipulation apparently is a successful interaction style when developing desktop multimedia using Macromedia Director while outperformed by command language when developing virtual reality applications?

At first sight Director and dvMockup may seem very much alike. Systematically comparing the two tools, however, reveals a number of fundamental differences.

5.1. Creating a Simple Application

Applications developed in Director or dvMockup typically consists of a large number of different elements such as graphics, 3D objects, sound/video files and scripts/behaviors. When creating a simple application these elements have to be put together as a coherent whole, which is then presented to the end-user. Director and dvMockup both provides means for organizing and putting together application elements for this purpose. The approaches enforced in the two tools are, however, fundamentally different in relation to both interface design and support for interaction

In Director, direct manipulation is used extensively when creating and modifying an application. Every element of the application (e.g. images or video) are represented graphically in the cast window (see figure 7) and can be used anywhere in the application by simply dragging them from the cast window on to the preview window (stage). This action creates a local instance of the element at that given point of the application. These instances can be modified (e.g. scaled or rotated) either in the preview window using direct manipulation or in the timeline (score) window using fill-in forms. In this way, multiple independent instances can be created and modified at multiple points of the application simply by using the mouse.
In the timeline (score) window the dynamic state of the application over time is represented graphically (see figure 7). The developer can interact with this representation and modify its properties using direct manipulation (dragging, dropping and scaling) as well as fill-in forms.

All phases of creating and modifying a simple application in Director are thus supported by various and coherent direct manipulation interaction styles. Due to the separation between cast and score, the developer furthermore only has to concentrate on the elements in use at one particular point of the application, and can leave the rest out of account.

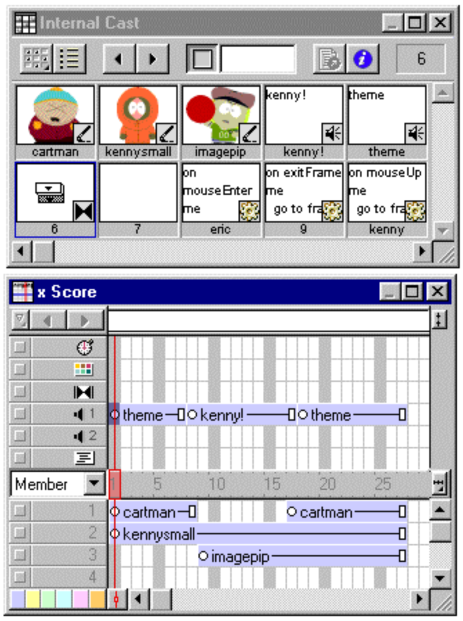


Figure 7. The Cast and Score Windows

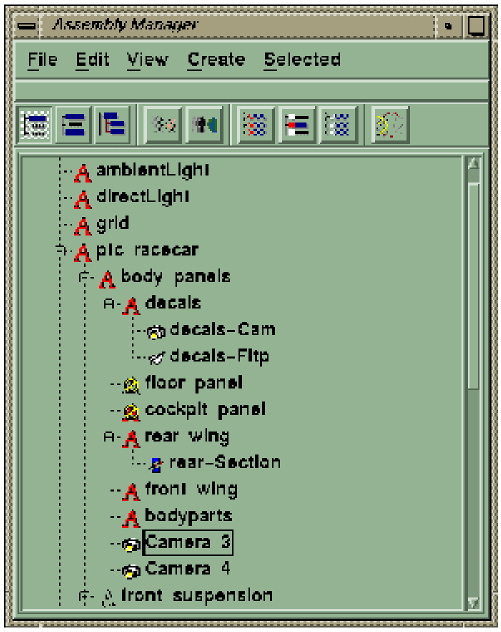


Figure 8. The Assembly Manager

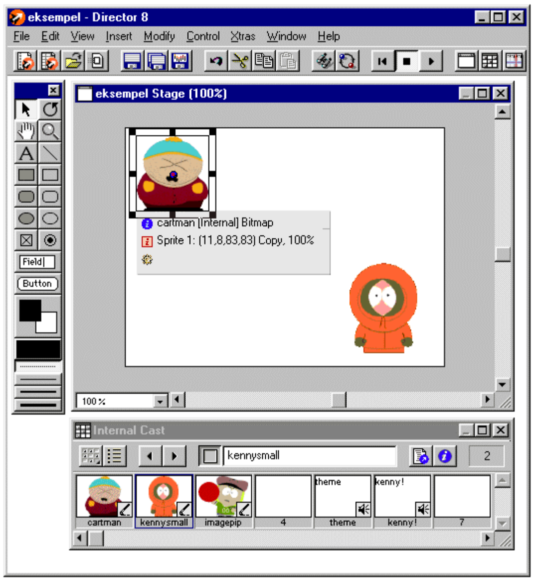
In dvMockup, direct manipulation is not used as extensively during the creation and modification of a simple application as in Director.
The elements of an application developed in dvMockup are grouped hierarchal within a “scene graph”, which can be accessed through the assembly manager (see figure 8). The structure of the scene graph can be compared to the structure of the file system on a computer. Every entry in the assembly manager corresponds to a unique element in the application with unique parameters. If the same 3D object is used twice in the application, it will, contrary to Director’s cast window, appear twice in the assembly manager. The scene graph facilitates manipulating whole “branches” of e.g. virtual 3D objects without affecting the mutual spatial relationship between the sub-objects

in the branch. This is very helpful when working with virtual 3D worlds, which can easily consist of more than a thousand 3D objects.
Manipulating the scene graph (e.g. moving an element from one branch to another or turning the visibility of an element on or off) is done by menu-selection. The developer cannot directly manipulate the layout of the scene graph by e.g. drag and drop.

Creating and modifying a simple application in dvMockup is thus supported by an interaction style, which unlike Director does not explore the potentials of direct manipulation much further than simple selection of objects and activation of buttons and menus. Whereas the cast window in Director can be considered a central placeholder with the state of the application at a given time reflected in the highly direct manipulation-able score, the assembly manager in dvMockup acts *both* as a placeholder *and* reflects the state of the application while at the same time supporting only a low level of direct manipulation. This makes the assembly manager approach in dvMockup less flexible than the cast/score approach in Director because *all elements* of the application has to be considered *at all times* of the application while at the same time having limited means for interaction as a developer. The lack of support for working with multiple instances of the same object in dvMockup furthermore contributes to making the scene graph more complex.

5.2. Previewing the Application

There are two fundamental differences between applications developed in Director and dvMockup. 1) Applications developed in Director are targeted at desktop 2D displays while applications developed in dvMockup are targeted at immersive 3D displays. 2) Applications developed in Director are *typically* explored screen-by-screen while applications developed in dvMockup constitute 3D worlds, which the user can explore freely.
These differences affects the previewing of the application as well as the potentials for direct manipulation in the preview window offered by the two tools.
In Director, the developer is *always* faced with a preview that matches *exactly* what the user will be presented with at a given time of the application (see figure 9). This makes previewing and directly manipulating the elements of the application accurate and non-problematic. In the preview of dvMockup, however, the developer can chose to see the virtual 3D world from *any* perspective wished, without relation to the perspective chosen by the end-user in the cave (see figure 10). The preview in Director furthermore matches the number of dimensions used when displaying the final application on a computer monitor while the 2D preview in dvMockup *does not match the 3D displaying of the virtual world in the cave.* The developer is thus left to *imagine* how the end-user may explore the application. This distances previewing from the user experience of the final application



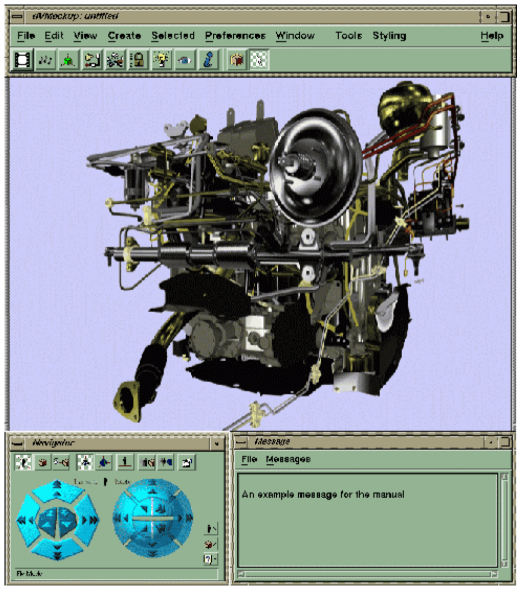


Figure 9. The Director interface

Figure 10. The dvMockup interface

Interaction with the preview constitutes yet another difference between the two tools. Whereas interaction with the preview in Director using mouse and keyboard matches the interaction with the final application, interacting with the preview of dvMockup does not. In the cave interaction is primarily a question of navigating a virtual world as well as selecting and manipulating virtual objects. This is typically supported using motion tracking and other 3D interaction devices. In the preview of dvMockup, however, the developer cannot interact as if being in the cave, but is limited to the use the mouse and keyboard or dedicated control panels in the interface which the end user will not have access to. This further extends the gap between the preview in dvMockup and the user experience of the final application and makes directly manipulating the elements in the preview window less direct.

5.3. Programming Functionality and Interaction

Multimedia and virtual reality applications typically have complex functionality and supports complex interaction. Tools for developing such applications must therefore support programming such functionality and interaction. In dvMockup, the user can create simple behaviors consistent of predefined events and actions and relate these to objects in the virtual 3D world. This is done by the use of menu-selection and fill-in forms (see figure 12). It is, however, not possible to extend the functionality of the predefined vents and actions in dvMockup by doing “real programming”. If additional functionality is desired, the application has to be hooked up to external executables programmed in e.g. C or C++.

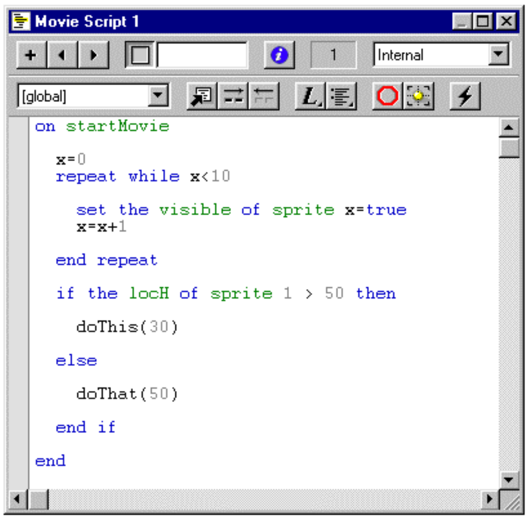


Figure 11. Scripting tool in Director

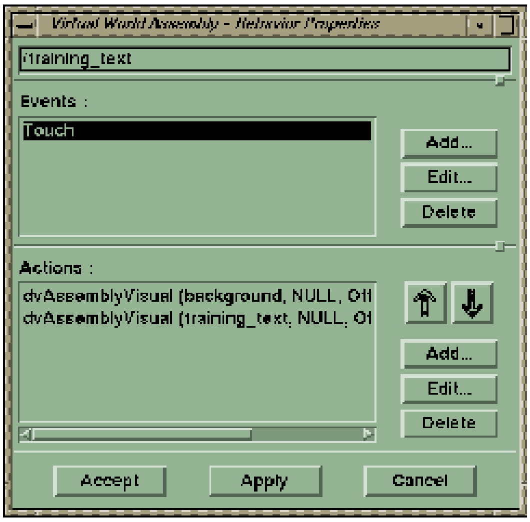


Figure 12. Behavior tool in dvMockup

While a menu-based tool for quick and easy creation of simple behaviors similar to the one in dvMockup is also accessible in Director, the latter furthermore provides a tool for command language interaction using high level scripting and object-orientation (see figure 11). Using Director, the developer benefits from a seamless transition between the scripting tool, the behavior tool and the graphical representation of the application. Observing Director in use by students and industrial developers clearly reveals a continuous iteration between these three tools at different phases of the development process. The support for advanced programming facilities in Director thus constitutes a significant difference between two tools.

The comparison of Director and dvMockup is summarized in table 1.

	Director	**dvMockup**
Creating a simple application	The elements of the application are *separated* from the state of the application at a given time.	The elements of the application *reflect* the state of the application at a given time.
Previewing the application	Previewing *matches* the end-user experience of and interaction with the application.	Previewing *does not match* the end-user experience and interaction.
Programming functionality and interaction	Use of both predefined events, actions, *and* object-oriented programming.	Exclusively use of predefined events and actions. *No direct support for programming.*

Table 1. Comparing Director and dvMockup

6. Conclusions

We have conducted a qualitative empirical study showing that implementing a virtual reality application using a command language tool and a direct manipulation tool required efforts in terms of time that are comparable. The command language tool, however, resulted in faster implementation during the most essential phases of the implementation process and thus outperforms the direct manipulation tool on a larger scale. The direct manipulation tool on the other hand resulted in fewer errors.

While the empirical results from comparing direct manipulation with command language interaction styles for development of virtual reality applications may suggest that within the virtual reality software development domain, command language is simply superior to direct manipulation, comparing dvMockup with Macromedia Director, which successfully employs direct manipulation, reveals a number of more specific issues, which may have negatively influenced the performance of dvMockup. These includes 1) the level at which the potentials of direct manipulation has been exploited, 2) the distance between development and target platform and 3) the support for switching between direct manipulation and a programming interface.

By focusing on application development for a six-sided cave using tools running on desktop computers we have, of course, taken things to the edge. There is a continuum of virtual reality displays for which the distance between development tool and target platform may not be as significant as for the six-sided cave.

A central question rises from this conclusion. Can direct manipulation be further exploited in tools for developing virtual reality applications? One thing is improving the user interface of development tools and further exploiting the potentials of direct manipulation with applications like Macromedia Director as role models, but how can we overcome the problem of distance between development and target platform? A relevant solution might be making direct manipulation more direct as discussed in Beaudouin-Lafon (2000) and Schkolne et al (2001). Developing virtual reality applications *in* the cave would be an interesting could be an interesting path for further exploration.

Acknowledgements

The authors would like to thank the development team: Mike H. Hougaard, Nikolaj Kolbe and Flemming N. Larsen. We also thank VR-MediaLab for access to virtual reality installations and development tools.

References

Beaudouin-Lafon M (2000) Instrumental Interaction: An Interaction Model for Designing Post-WIMP User Interfaces. CHI Letters 3(1):446-453

Benbasat I, Todd, P (1993) An Experimental Investigation of Interface Design Alternatives: Icon vs. Text and Direct Manipulation vs. Menus. International Journal of Man-Machine Studies 38:369-402

Hougaard MH, Kolbe N, Larsen FN (2001) Comparison of Tools for Developing Virtual Reality Application (in Danish), Intermedia, Aalborg University

Jepsen LO, Mathiassen L, Nielsen PA (1989) Back to Thinking Mode: Diaries for the Management of Information System Development Projects. Behaviour and Information Technology 8(3):207-217

Kjeldskov J (2001) Combining Interaction Techniques And Display Types For Virtual Reality. In: Smith W, Thomas R, Apperley M (eds.) Proceedings of OzCHI 2001, Edith Cowan University Press, Churchlands, Australia, pp 77-83

Margono S, Shneiderman B (1987) A Study of File Manipulation by Novices Using Commands vs. Direct Manipulation. In: Proceedings of the 26th Annual Technical Symposium, ACM, Washington, pp 57-62

Morgan K, Morris RL, Gibbs S (1991) When Does a Mouse Become a Rat? or … Comparing the Performance and Preferences in Direct Manipulation and Command Line Environment. Computer Journal 34:265-271.

Naur P (1983) Program Development Studies Based on Diaries. In: Green TR et al. (eds) Psychology of Computer Use, Academic Press, London, pp 159-170

Schkolne S, Pruett M, Schröder P (2001) Surface Drawing: Creating Organic 3D Shapes with the Hand and Tangible Tools. CHI Letters 2(1):261-268.

Shneiderman B (1998) Designing the User Interface: Strategies for Effective Human-Computer Interaction. Addison Wesley, Longman, Reading, Massachussetts

Published in *Proceedings of the 9th IFIP TC13 Conference on Human Computer Interaction, Interact 2003* (Zürich, Switzerland, September 2003). Zürich, IOS Press

The Process of Developing a Mobile Device for Communication in a Safety-Critical Domain

Jesper Kjeldskov

Department of Information Systems
University of Melbourne
Parkville
Victoria 3010
Melbourne, Australia

jesper@cs.auc.dk

Jan Stage

Department of Management Science and Information Systems
University of Auckland
Private Bag 92019
Auckland, New Zealand

jans@cs.auc.dk

Abstract: This paper describes the process of developing an interactive mobile system for use in a safety-critical domain. The system was developed to inquire into the extent of which coordination between people carrying out a collaborative work task can be improved by means of computer mediated communication on a mobile device. The application area for the device was controlling and monitoring processes in the safety-critical domain of a large container vessel when it arrives at or leaves a harbor. The development process involved ethnographic studies of work activities in this domain, analysis of communication between actors involved in the work activities, design and implementation of an experimental prototype, and two usability evaluations of the prototype. We emphasize the analysis and design activities and outline the challenges involved in the development of the mobile device. Also, we outline how our experiences are relevant in a more general context.

Keywords: Mobile device, development process, analysis and design, usability evaluation, safety-critical domain, computer mediated communication

1 Introduction

Mobile and wearable computer devices are being developed for use in a broad variety of application areas. Recent research has devoted much attention to the extent to which distributed mobile users can benefit from mobile computer systems. This includes research on situations where actors are concerned with computerized information and processes of critical importance remote from their current location. Examples count distributed process control and error diagnosing in wastewater treatment plants (Nielsen and Søndergaard, 2000) and the use of mobile multimedia communication for telemedicine and early diagnosing in emergency ambulance services (van den Anker and Lichtveld, 2000). Safety-critical domains are potentially interesting application areas of mobile computers. A safety-critical computer system can be defined as a *"computer electronic or electromechanical system whose failure may cause injury or death to human beings"* (Palanque et al., 1998). In relation to human-computer interaction in safety-critical domains, the key problem is not software failure as such but rather failure to support correct user interaction caused by poor interface design. Examples of this count the British Midland 1989 air crash, in interface design in the cockpit contributed to pilots erroneously shutting down the only operational (Ladkin, 1998) or the Three Mile Island nuclear power plant accident in 1979. Based on these and similar experiences it has been suggested that instead of forcing additional rules and regulations on the operators of complex computerized industrial installations, better designed user interfaces could support the avoidance of hazardous situations (Fields et al., 1999; Leveson, 1995; Mackay, 1999). This also applies to the use of mobile computer systems is such domains (Nielsen and Søndergaard, 2000; van den Anker and Lichtveld, 2000).

Analysis, design, and prototyping methods are general means for supporting development of user interfaces and software systems. Generally very little has been published on the processes of developing mobile device applications. Interesting examples count Sharples et al. (2002), Mikkonen et al. (2002) and Iacucci et al. (2000) who outline a number of differences from the development of traditional software. In relation to *safety critical* mobile device applications Knight (2002) touches upon challenges in system development introduced by mobile computers and Bertelsen and Nielsen (2000) presents a design tool that could be applied for supporting aspects of these. Furthermore, Navarre et al. (2003) present a design framework applicable to stationary as well as mobile safety critical interactive systems.

Experience from processes of developing mobile systems for safety-critical domains can support other similar development processes. Experience always originates from a particular situation and therefore cannot replace general methods. Nevertheless, when general methods have not yet been developed, systematically collected experience can be a valuable substitute. Furthermore, such experiences are often needed to support the development of improved analysis and design methods for a specific domain of application.

The following sections presents experiences from the process of developing a mobile device application for supporting communication in a safety-critical domain. Section 2 describes the basic requirements for the device and provides an outline of the overall development process. Section 3 presents the application domain analysis conducted, involving ethnographic studies of work activities in the safety-critical domain of focus. In section 4, we describe the problem domain analysis focusing on communication between actors involved in the work activities observed, resulting in an information-architecture for the mobile device. Section 5 presents the design and implementation of an experimental prototype. Section 6 describes two usability evaluations of the prototype.

As focus of this paper is on the activities conducted during the development process, the challenges involved, and the results achieved, the actual design produced will only be presented in order to illustrate aspects of these. Finally, section 7 concludes the presentation and discusses how the experiences presented are relevant in a more general context.

2 Development Process

The point of departure of the development process was an essential communication problem experienced in the safety-critical domain of operating a large container vessel during arrival at and departure from a harbor. This operation requires communication between actors that are physically distributed on the vessel. Currently, this communication is based on spoken natural language being transmitted through handheld VHF radios.

The fundamental design idea was to replace the radios with a mobile device that facilitates exchange of predefined text messages. This is similar to the way in which SMS and e-mail applications on mobile phones and PDAs have been employed to replace direct spoken communication with network-based exchange of text messages. When communicating by means of textual messages on a mobile device, communication is no longer subject to the ephemeral nature of spoken utterances. Moreover, it can be conducted asynchronously, it is not influenced by surrounding noise, and information can be integrated with other computer-based data.

In order to explore this fundamental idea, we developed a prototype of such a mobile device. The activities of this development process are illustrated in figure 1.

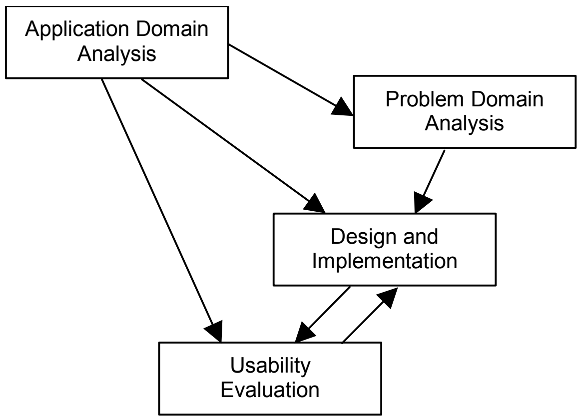


Figure 1: The activities of the development process

The first activity was analysis of the application domain. The basic approach was to conduct ethnographic studies of work activities in the safety-critical domain. The second activity was the problem domain analysis that focused on the communication between actors involved in the work activities and resulted in an information architecture for the mobile device. The third activity was to design and implement an experimental prototype. The fourth activity was usability evaluations of the prototype.

3 Application Domain Analysis

The development of the mobile communication device was carried out in collaboration with Maersk-Sealand. It was based on ethnographic field studies, with contextual interviews and video analysis.

3.1 Field Study

Maersk-Sealand operates some of the world's largest container vessels of sizes equivalent to 3½ soccer fields. A sketch of a ship from this class is shown in figure 2.

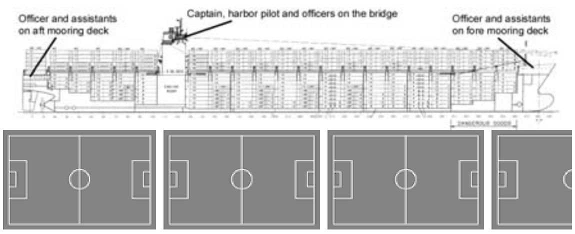


Figure 2: Sketch of Sally Maersk compared to the size of soccer fields

The operation of such a vessel is a safety-critical domain. Especially when maneuvering inside a harbor, erroneous actions may result in the vessel running aground, into the quay, or colliding with other ships. In either case, such collisions would cause serious material damage, potentially severe injuries on personnel, and possible loss of human life.

When the ship is ready for departure, the first step in leaving the quay is to let go of the mooring lines that are holding it in position. A configuration of aft mooring lines is shown in figure 3.

Figure 3: The aft mooring lines of Sally Maersk

When the crew lets a line go, it will remain in the water for a period of time during which no means of propulsion is available due to the risk of lines getting sucked in and wrapped around a propeller or thruster. During this time, the vessel can only maneuvered by means of the remaining lines.

Due to the huge size of the vessel, the work tasks involved when letting go the lines are distributed among a number of actors located at strategic positions, as annotated at the top of figure 2. On the bridge, the captain and other personnel control the rudder, propeller and thrusters. Fore and aft, the first and second officers control the winches for heaving in the lines.

To ensure the safety of the operation, individual work tasks are carefully coordinated and carried out under strict command of the captain in charge. At present the coordination is primarily based on oral communication via VHF radios.

In order to carry out the operation of departure in a safe manner, the captain needs an overview and total control over the propulsion, direction and mooring of the ship. At present much of this only exists as a mental model in the head of the captain based on his perception of the ongoing communication between bridge and deck. Therefore, considerable cognitive resources are spent on establishing and maintaining *common ground* (Clark and Schaefer, 1989) among the cooperating actors.

The field studies were documented in qualitative descriptions of the application domain (Andersen, 2000; Nielsen, 2000). In addition, the processes of departing from and arriving at harbor were recorded on video capturing overall views of the captain, harbor pilot and officers on the bridge as well as close-up views of the interaction with key instruments. The audio channel captured inter-personal communication on the bridge and VHF radio communication with the crews fore and aft.

3.2 Video Analysis

In order to facilitate systematic analysis, a person with detailed insight into the application domain transcribed a selection of the video recordings. The transcriptions amount to approximately 200 pages.

Through analysis of recordings, transcriptions, and follow-up interviews, we uncovered the following communication problems:

- Sound quality is often poor
- Utterances are not persistent
- Communication is time consuming
- There is a language barrier
- The bridge is a bottleneck
- Information is not integrated with other systems

In the maritime domain, spoken communication is prevalent and often carries the majority of vital information. To handle this, a well-established set of formalized procedures for communication exists. However, as the size of vessels and the use of technology increase so does the complexity of systems and the cognitive overhead and amount of communication required for operating the systems. Thus a strong motivation exists for supporting and improving communication.

4 Problem Domain Analysis

The problem domain analysis was directed towards design of a mobile device. It was based on an object-oriented analysis method (Mathiassen et al., 2000) and has similarities to the work presented by Navarre et al. (2003).

4.1 Objects and Classes

The object-oriented analysis facilitated identification of relevant objects and their classes. The basis for this was interviews, video recordings, and transcripts. From the interviews and video recordings we identified relevant actors. From the transcripts we identified the key conversation objects as in the following extract from the transcript:

```
1  <Captain>      you can let go the bow line
2  <1st officer>  let go bow line
3  <Captain>      you can take the stern spring
4  <2nd officer>  letting go stern spring
5  <1st officer>  bow line let go
6  <Captain>      bow line let go
7  <2nd officer>  and stern spring let go
8  <Captain>      stern spring let go
9  <Captain>      just let go the stern line also
10 <2nd officer>  let go line aft
11 <1st officer>  and we have the bow line home
12 <Captain>      ok
13 <2nd officer>  and all let go aft
14 <Captain>      all let go aft
```

This conversation is about three interweaved Let go tasks. The tasks are conducted by two different Teams, represented by the 1st and 2nd officer. They involve two locations: the bow and the stern. The classes and relations are illustrated in figure 4.

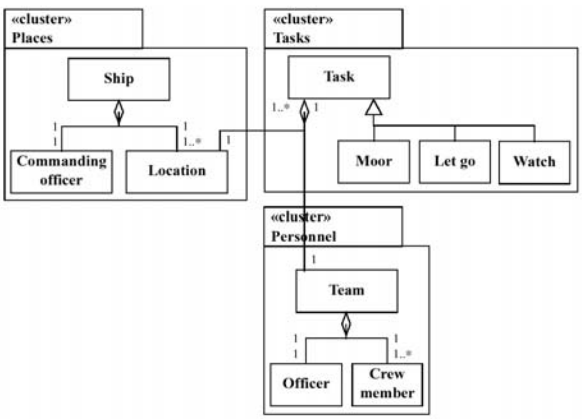


Figure 4: Class diagram of the problem domain

Communication consisting of a number of interweaved tracks of conversations can be difficult to overview when sorted by sequence of utterance. The transcription extract above illustrates this, as it is actually an integration of three conversational tracks taking place in parallel.

Grouping text in accordance to object rather than sequence enables the creation of a more comprehensible representation of communication threads. The objects of this representation are simply expressed in the class diagram.

4.2 Object Behaviour

The object-oriented analysis also involves description of behaviour for the classes listed in the class diagram. The most interesting classes were those related to the different tasks. State charts for these classes were used to model sequences of communications as the ones illustrated by the transcripts above and sequences of sub-tasks being carried out sequentially and in parallel.

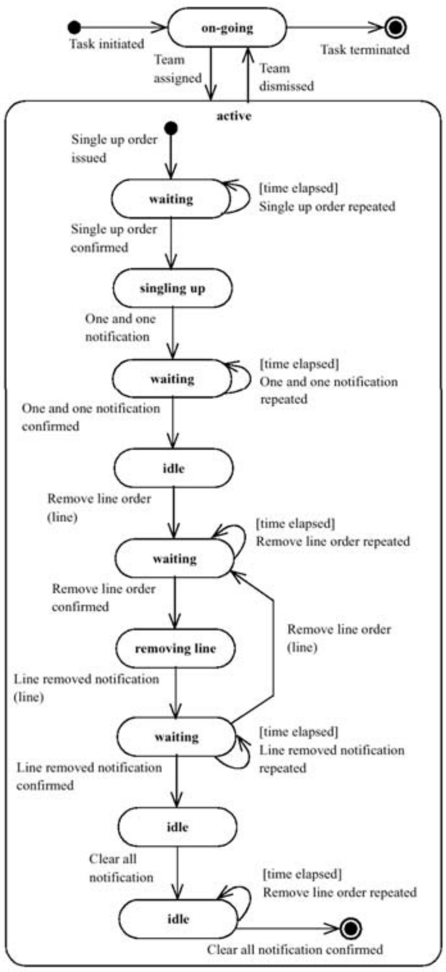


Figure 5: State chart diagram for the Let go task

5 Design and Implementation

A prototype was designed and implemented on the basis of the results from the analysis activities. This prototype should support the task of letting go the lines.

The prototype was targeted at a Compaq iPAQ 3630 handheld computer. Apart from a touch screen, this device facilitated interaction by means of a five-way key located below the display, suitable for one-handed interaction. Due to the potentially harsh conditions of use, in which pen-based interaction might be problematic, we decided to design all interaction for this key.

The configuration of the system consisted of three iPAQs connected through a wireless network. One device was intended for the captain on the bridge while the other two were for the 1st and 2nd officers on the fore and aft deck respectively.

5.1 Overall Design

The overall design was based on three key ideas. The first was to replace verbal communication with exchange of predefined text messages. Asynchronous text-based messaging is a flexible, ubiquitous and persistent communication channel that requires only low cognitive overhead (Churchill and Bly, 1999; Popolov et al., 2000). Thus we expected shifting to text-based communication on mobile devices could eliminate or reduce some of the problems that were listed in section 3.2.

The second idea was to provide an updated representation of the process that is monitored and controlled on different levels of abstraction. In order to avoid accidents in safety-critical domains, it is critical to understand the state of the system being operated. Rasmussen (1983; 1986) suggests that computer interfaces should be designed to improve operators' reasoning about the domain of operation and thus support human interaction contrary of total system automation as also discussed by Norman (1990). For this purpose, Lind (1994; 1999) suggests a formalism for representing complex systems or environments as they were intentionally designed.

The third idea was to use the structure of conversations as the means for interacting with the user. On an overall level, a conversation can be categorized by aspect and tense (Andersen, 2000), so it is either:

- Imminent (future tense)
- Executing (present tense)
- Ended (past tense)

While *executing* (present) conversations are still open for negotiation. *Ended* (past) conversations imply some kind of mutual agreement having been made among the communicating parties. *Imminent* (future) conversations are characterized by potentially being initiated when and if appropriate in relation to preceding conversations (ended and executing). Knowing the state of executing conversations, this can be represented visually for fast access: has a request been met? Has an agreement been made? Etc. Also, possible future utterances may be deduced and prioritized over others in the form of predefined standard-phrases as seen on some SMS-enabled mobile phones. Thus demands for user-interaction may be reduced.

5.2 User Interface

The user interface is divided into four sections:

- Pictogram of ship and mooring (past and present)
- List of completed communication threads (past)
- List of ongoing communication threads (present)
- List of unexecuted commands (future)

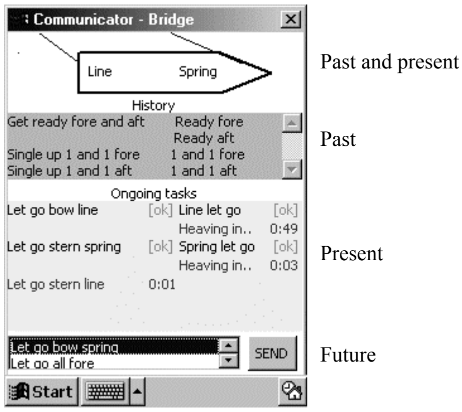


Figure 6: The interface on the bridge

The user interface for the bridge is illustrated in figure 6. At the bottom of the screen there is a list of unexecuted commands and confirmations. The order of the list corresponds to the standard sequence of the overall operation, and possible utterances only appear when appropriate in relation to the state of the task (figure 5). The state chart diagrams were used to design these sequences. By default, the most likely next step of the operation is highlighted.

The most important element of the interface is the list of ongoing tasks. When a command is executed, it appears on the list of ongoing threads of communication representing uncompleted tasks. Next to it, a counter displays the time passed while waiting for confirmation (figure 7a). When a command is confirmed the timer is substituted by the text "[ok]" followed by a description of the current activity (e.g. "Singling up..."). A counter next to this displays the time passed since confirmation (figure 7b). When a task is reported completed, a short statement (e.g. "1 and 1 fore") substitutes the description of activity and the captain is prompted for confirmation (figure 7c). When the completion of a task is confirmed), this is indicated by the text "[ok]" (figure 7d).

Figure 7: Stages of a task

When the captain confirms the completion of a task, the corresponding thread of communication is removed from the list of ongoing tasks and added at the bottom of the history list. When the history list is full, it automatically scrolls the oldest commands and statements out of immediate sight.

For quick reference and supporting different level of abstractions in the interaction with the system, a simple pictogram at the top of the screen graphically represents the lines attached to the quay and textually shows the current status of fore and aft mooring.

On deck, the interface is very similar to that on the bridge thus providing the first and second officers with a view on the present status of the mooring and a list of all past and ongoing communication among the distributed actors. In the list of ongoing tasks, however, officers on deck are requested to confirm commands executed by the captain such as "Let go bow spring". Correspondingly, the list of pre-defined commands only contains those appropriate at the specific location.

5.3 Implementation

The system was implemented using Microsoft Embedded Visual Basic. For the first prototype we connected the iPAQ computers through an IEEE 802.11 11Mb wireless TCP/IP network. Given the specific use domain this will probably not be a viable solution for the final application due to limited range, potential radio interference and consequential risk of loosing network packages. Furthermore TCP/IP has limited capabilities for parallel data and voice transmissions. Instead the use of other network technologies like e.g. Tetra Net (voice and data) should be investigated.

The application running on the captain's device worked as a server containing a formalized representation of the communication pattern of the task. The devices on deck logged on to this server and identified their physical location. During operation, function calls and unique command identifiers were exchanged over the wireless network. All network communication was broadcast but processed and represented differently on each device in accordance to their physical location (bridge, fore or aft) and the desired language (defined in an external text-file on each device).

6 Usability Evaluation

Evaluating the usability of the prototype was a challenge. Firstly, the use of the device is closely related to specific work activities in a specialized physical context, which can be difficult to recreate realistically in a usability laboratory. Conducting evaluations in the field would, on the other hand, limit means of control and complicate data collection. Furthermore, field evaluations of an early stage prototype in a safety-critical domain could cause a hazardous situation. Hence we decided to carry out two different evaluations in controlled settings.

6.1 Heuristic Inspection

In our first evaluation, we applied an established method for expert inspection developed by Nielsen and Molich (1990). The aim of this approach is to test the basic design of an interface using few resources and without involving users.

Figure 8: Heuristic inspection

Three trained usability experts were given a 15-minute joint oral introduction to the application domain of the prototype. Aided by a standard heuristic for usability design (Dix, 1998, p. 413) each person spent one hour checking for usability problems while using the prototype. Following the inspections, the team was given one hour of discussion during which a final list of problems should be produced.

The inspection setup consisted of two Compaq iPAQs and a PocketPC simulator on a laptop PC connected through a wireless network. The iPAQs displayed the interfaces for officers fore and aft respectively while the laptop displayed the interface for the captain on the bridge. Two A4 handouts depicted standard patterns of mooring and explained 10 basic concepts and notions of the maritime context for quick reference. All three evaluators were able to use the prototype on their own. In total, 27 usability problems were identified, primarily concerning the graphical interface design.

6.2 Use in a Simulated Context

Our second evaluation focused on the performance of the prototype in the hands of prospective users in a more realistic work setting. For this study we used a state-of-the-art ship simulator facilitating a fully equipped bridge and a high-fidelity interactive scenario of the operation of a large vessel. Three teams of two trained maritime officers were given the overall task of departing from harbor using the prototype for communication between bridge and deck. One test subject acted as captain on the simulated bridge while the other acted as 1st officer on the fore deck in a neighboring room. For simplicity, commands targeted at the 2nd officer on the aft deck were fed directly into the simulation and feedback was given by the simulation operator.

The simulator was set up to imitate the operation of a large vessel in challenging weather and traffic conditions corresponding to a real world situation observed during our field studies (Nielsen, 2000).

Figure 9: Video documentation of use in a ship simulator

During the evaluation, the test subjects were asked to think-aloud, explaining their actions and their use of the prototype. Two evaluators located on the bridge and deck respectively observed the test subjects and asked questions for clarification. Total views of the bridge, deck, simulator control room as well as close-up views of the mobile devices were captured by four video cameras and merged into one high-quality video signal providing a synchronized view of the whole setup (figure 9). Following the evaluation, a group interview of 10-15 minutes was carried out.

Observing the use of the prototype by prospective users performing a real work task in the simulator provided rich data on the usability of the design. The study revealed 22 usability problems experienced by more than one user.

6.3 Comparison

The two evaluations provided substantial input for redesign. Some of the problems listed in section 3.2 were reduced, but other problems emerged.

The quantitative results produced by the two evaluations were very different. When merging the results, a total of 43 problems were identified of which 63% were accounted for by the expert evaluators. However, real users never experienced 78% of the problems revealed in the inspection, thus questioning if they are problems at all. This indicates that although expert inspections may reveal a large number of usability problems, their importance and scope can be difficult to assess and compare without access to richer data.

To the contrary, all problems identified in the simulator were much more detailed and grounded. Balancing resources, however, the simulator study required 40 man-hours while the inspection required only 10.

7 Conclusion

The sections above describe how a mobile device supporting a collaborative work activity in a specific safety-critical domain was developed. The development process involved ethnographic field studies, application domain analysis, problem domain analysis, design and implementation, and usability evaluation.

The safety-critical domain considered was found to be characterized by established procedures and highly formalized communication. This facilitated the development of useful models of the problem domain and the development of a novel design supporting existing work practice and overcoming a number of present limitations in communication.

Thus the process applied to the development of the presented prototype proved very useful in terms of assisting the identification and modelling of relevant characteristics and structures as well as providing input directly applicable in interface design. While we believe that the described process has general value for the development of similar systems, this should, however, be evaluated by applying it to other cases of developing mobile device applications for specialized domains.

Acknowledgements

The authors thank the evaluation team and test subjects, Svendborg International Maritime Academy and CHMI. The evaluations were planned, conducted and analyzed in cooperation with Mikael B. Skov

References

Andersen, P. B. (2000), Communication and work on maritime bridges. CHMI Research report CHMI-1-2000 http://www.cs.auc.dk/~pba/ElasticSystems

Bertelsen, O. and Nielsen, C. (2000) Augmented Reality as a Design Tool for Mobile Devices, in *Proceedings of DIS 2000,* ACM.

van den Anker, F. W. G. and Lichtveld, R. A. (2000), Early Evaluation of New Technologies: The Case of Mobile Multimedia Communications For Emergency

Medicine, in Vincent C. and de Mal B (eds.) *Safety in Medicine*, Qxford: Elsewier Science.

Churchill, E. F. and Bly, S. (1999), It's all in the words: Supporting work activities with lightweight tools, in *Proceedings of ACM Siggroup'99*, pp. 40-49.

Clark, H.H. and Schaefer, E. F. (1989), Contributing to discourse, *Cognitive Science*, 1989(13), 259-294.

Dix, A. et al. (1998), *Human-Computer Interaction – Second Edition,* London, Prentice Hall Europe.

Fields, R. Paterno et al. (1999), Comparing Design and Options for Allocating Communication Media in Cooperative Safety-Critical Contexts: A Method and a Case Study, *ACM TOCHI* 6(4), 370-398.

Iacucci, G., Kuutti, K. and Ranta, M. (2000) On the Move with a Magic Thing: Role Playing in Concept Design of Mobile Services and Devices, in *Proceedings of DIS 2000,* ACM.

Knight, J. C. (2002) Safety Critical Systems: Challenges and Directions, in *Proceedings of ICSE'02,* Orlando, Florida, ACM.

Ladkin, P. B. (1998), Computer-Related Incidents with Commercial Aircrafts. University of Bielefeld, http://www.rvs.uni-bielefeld.de/publications /Incidents/

Leveson, N. G. (1995), *Safeware: System Safety and Computers,* Addison-Wesley Longman Publ. Co., Inc., Reading, MA.

Lind, M. (1994), Modeling Goals and Functions in Complex Industrial Plants, *Applied Artificial Intelligence*, 8(2), 259-283.

Lind, M. (1999), Plant Modeling for Human Supervisory Control, *Transactions of the Institution of Measurement and Control*, 21(4/5), 171-180.

Mackay, W. E. (1999). Is Paper Safer? The role of Paper Flight Strips in Air Traffic Control, *ACM TOCHI* 6(4), 311-340.

Mathiassen, L., Munk-Madsen, A., Nielsen, P. A., and Stage, J. (2000), *Object-Oriented Analysis and Design.* Aalborg, Marko.

Mikkonen, M., Vayrynen, S., Ikonen, V. and Heikkila, O. (2002) User and Concept Studies as Tools in Developing Mobile Communication Services, *Personal and Ubiquitous Computing*, 2002(6), 113-124, Springer-Verlag.

Navarre, D. Palanque, P. and Bastide, R. (2003) A Tool-Supported Design Framework for Safety Critical Interactive Systems. Interacting with computers, Elsevier, to appear 2003.

Nielsen, C. and Søndergaard, A. (2000), Designing for mobility - an integration approach supporting multiple technologies, in *Proceedings of NordiCHI 2000,* 23-25 October 2000, Royal Institute of Technology, Stockholm, Sweden.

Nielsen, J. and Molich, R. (1990), Heuristic evaluation of user interfaces, in *Proceedings of CHI'90,* Seattle, USA, 249 - 256, ACM Press.

Nielsen, M. (2000), Letting go the lines: Departure from Felixstowe harbor. CHMI Research Report CHMI-4-2000 http://www.cs.auc.dk/~pba/ElasticSystems.

Norman, D. (1990), The 'Problem' With Automation: Inappropriate Feedback And Interaction Not Over-automation, in Broadbent D.E. et al. (eds.) *Human Factors In Hazardous Situations,* Oxford, Clarendon Press, 137-145.

Palanque, P. Paterno, F. and Wright, P. (1998), Designing User Interfaces for Safety-critical Systems, in Proceedings of CHI 98, Los Angeles, USA, ACM Press.

Popolov, D. Callaghan, M. and Luker, P. (2000), Conversation Space: Visualising Multi-threaded Conversation, in *Proceedings of AVI2000,* Palermo, Italy, ACM, 246-249.

Rasmussen, J. (1983), Skills, Rules and Knowledge: Signals, Signs and Symbols and Other Distinctions in Human Performance Models, *IEEE Transactions on Systems, man and Cybernetics* 13(3).

Rasmussen, J. (1986), *Information Processing and Human-Machine Interaction.* New York, North-Holland.

Sharples, M, Corlett, D. and Westmancott, O. (2002) The Design and Implementation of a Mobile Learning Resource, *Personal and Ubiquitous Computing* 2002(6), 220-234, Springer-Verlag.

Appendix B

Research Classification

B.1 Classification of HCI Research Papers for Virtual Reality

Research Method

Research Purpose

	Case studies	Field studies	Action research	Lab experiment	Survey research	Applied research	Basic research	Normative writings
Understand			2	29, 33, 40, 52, 62			53	
Engineer		10, 27				1, 3, 5, 6, 7, 9, 11, 12, 13, 14, 15, 16, 18, 21, 23, 24, 26, 31, 34, 35, 37, 38, 41, 46, 47, 48, 50, 51, 57, 61, 63, 65	39, 43, 58, 46	
Re-engineer						19, 21		
Evaluate		10, 25, 51		3, 5, 8, 12, 15, 20, 21, 22, 24, 26, 30, 32, 34, 38, 39, 42, 44, 46, 47, 49, 50, 54, 56, 59	33			
Describe				36, 45		1, 11, 13, 17, 20, 28, 64	23, 55	49, 56, 60

Table 1. Classification of HCI research papers for virtual reality

1. Ahn S. C., Kim I., Kim H., Kwon Y. and Ko H. (2001) Audience Interaction for Virtual Reality Theatre and its Implementation. In Proceedings of VRST 2001, ACM
2. Allison D. and Hodges L. F. (2000) Virtual Reality for Education? In Proceedings of VRST 2000, Seoul, Korea, ACM
3. Ansar A., Rodrigues D., Desai J. P., Daniilidis K., Kumar V. and Campos M. F. M. (2001) Visual and haptic collaborative tele-presence. Computers and Graphics (2001) Vol. 25, 789-798
4. Anstey J., Pape D. and Sandin D. (2000) The Thing Growing: Autonomous Characters in Virtual Reality Interactive Fiction. In Proceedings of IEEE VR 2000
5. Asai K., Osawa N., Takahashi H. and Sugimoto Y. Y. (2000) Eye Mark Pointer in Immersive Projection Display. In Proceedings of IEEE VR 2000
6. Bimber O., Encarnacao L. M. and Schmalstieg D. (2000) Real Mirrors Reflecting Virtual Worlds. In Proceedings of IEEE VR 2000
7. Bowman D. A., Wingrave C. A., Campbell J. M., Ly V. Q. and Rhoton C. J. (2002) Novel Uses of Pinch Gloces for Virtual Environment Interaction Techniques. Virtual Reality (2002) (6), 122-129, Springer-Verlag
8. Bowman D. A. and Wingrave C. A. (2001) Design and Evaluation of Menu Systems for Immersive Virtual Environments. In Proceedings of IEEE VR 2001
9. Chen H. and Sun H. (2002) Real-time Haptic Sculpturing in Virtual Volume Space. In Proceedings of VRST 2002, Hong Kong, ACM

10. Crabtree A., Rodden T. and Mariani J. (2002) Designing Virtual Environments to Support Cooperation in the Real World. Virtual Reality (2002) (6), 63-74, Springer-Verlag

11. de Haan G., Koutek M. and Post F. H. (2002) Towards Intuitive Exploration Tools for Data Visualization in VR. In Proceedings of VRST 2002, Hong Kong, ACM

12. Duchowski A. T., Medlin E., Gramopadhye A., Melloy B. and Nair S. (2001) Binocular Eye Tracking in VR for Visual Inspection Training. In Proceedings of VRST 2001, ACM

13. Ehnes J., Knopfle C. and Unbescheiden M. (2001) The Pen and Paper Paradigm – Supporting Multiple Users on the Virtual Table. In Proceedings of IEEE VR 2001

14. Foskey M., Otaduy M. A. and Lin M. C. (2002) ArtNova: Touch-Enabled 3D Model Design. In Proceedings of IEEE VR 2002.

15. Frohlich B. and Plate J. (2000) The Cubic Mouse: A New Device for Three-Dimensional Input. In Proceedings of CHI 2000, ACM

16. Frolich B., Beers A. and Baraff D. (2000) Physically-Based Manipulation on the Resonsive Workbanch. In Proceedings of IEEE VR 2000

17. Fuhrmann A. L., Prikryl J., Tobler. And Purgathofer W. (2001) Interactive Content for Presentations in Virtual Reality. In Proceedings of VRST 2001, ACM

18. Gregory A. D., Ehmann S. A. and Lin M. C. (2000) inTouch: Interactive Multiresolution Modelling and 3D Painting with a haptic Interface. In Proceedings of IEEE VR 2000

19. Grossman T., Balakrishnan R., Kurtenbach G., Fitzmaurice G, Kahn A. and Buxton B. (2001) Interaction Techniques for 3D Modeling on Large Displays. In Proceedings of I3D 2001, ACM

20. Harders M. and Szzekely G. (2002) Improving Medical Segmentation with Haptic Interaction. In Proceedings of IEEE VR 2002

21. Ikei Y., Yamada M. and Fukuda S. (2001) A New Design of Haptic Texture Display – Texture Display2 – and its Preliminary Evaluation. In Proceedings of IEEE VR 2001

22. Iwata H., Yano H. and Nakaizumi F. (2001) Gait Master: A Versatile Locomotion Interface for Uneven Virtual Terrain. In Proceedings of IEEE VR 2001

23. Jiang H., Kessler D. and Nonnemaker J. (2002) DEMIS: A Dynamic Event Model for Interactive Systems. In Proceedings of VRST 2002, Hong Kong, ACM

24. Kakusho K., Kitawaki J., Hagihara S. and Minoh M. (2000) Adjusting the Difference between 3D Spaces in VR Systems and Human Perception through Object manipulation. In Proceedings of IEEE VR 2000

25. Keefe D. F., Feliz D. A., Moscovich T. Laidlaw D. H. and LaViola Jr. J. J. (2001) CavePainting: A Fully Immersive 3D Artistic Medium and Interactive Experience. In Proceedings of I3D 2001, ACM

26. Kitamura Y., Otaga S. and Kishino F (2002) A Manipulation Environment of Virtual and Real Objects using a Magnetic Metaphor. In Proceedings of VRST 2002, Hong Kong, ACM

27. Knopfle C. and Voss G. (2000) An Intuitive VR User Interface for Design Review. In Proceedings of AVI 2000, ACM

28. LaViola Jr. J. J., Feliz D. A., Keefe D. F. and Zeleznik R. C. (2001) Hands-Free Multi-Scale Navigation in Virtual Environments. In Proceedings of I3D 2001, ACM

29. Lecuyer A., Burkhardt J., Coquillart S. and Coiffet P. (2001) “Boundary of Illusion”: an Experiment of Sensory Integration with a Pseudo-Haptic System. In Proceedings of IEEE VR 2001

30. Lecuyer A., Coquillard S. and Kheddar A. (2000) Pseudo-Haptic Feedback: Can Isometric Input Devices Simulate Force Feedback? In Proceedings of IEEE VR 2000

31. Leibe B., Starner T., Ribarsky W., Wartell Z., Krum D., Singletary B. and Hodges L. (2000) The Perceptive Workbench: Toward Spontaneous and Natural Interaction in Semi Immersive Virtual Environments. In Proceedings of IEEE VR 2000

32. Li T. and Ting H. (2000) An Intelligent User Interface with Motion Planning for 3D Navigation. In Proceedings of IEEE VR 2000

33. Lin C. R., Loftin R. B. and Nelson Jr. E. R. (2000) Interaction with Geoscience Data in an Immersive Environment. In Proceedings of IEEE VR 2000

34. Lin R. R. and Loftin R. B. (2000) VR User Interface: Closed World Interaction. In Proceedings of VRST 2000, Seoul, Korea, ACM

35. Lindeman R. W., Templeman J. N., Sibers J. L. and Cutler J. R. (2002) Handling of Virtual Contact in Immersive Virtual Environments: Beyond Visuals. Virtual Reality (2002) (6), 130-139, Springer-Verlag

36. Lindeman R. W., Sibert J. L. and Templeman J. N. (2001) The Effect of 3D Widget Representation and Simulated Surface Constraints on Interaction in Virtual Environments. In Proceedings of IEEE VR 2001

37. McDonnell K. T., Qin H. and Wlodarczyk R. A. (2001) Virtual Clay: A Real-time Schulpting System with Haptic Toolkits. In Proceedings of I3D 2001, ACM

38. Mulder J. D. and van Liere R. (2000) Enhancing Fish Tank VR. In Proceedings of IEEE VR 2000

39. Neale H., Cobb S. and Wilson J. (2002) A Front-Ended Approach to the User-Centered Design of VEs. In Proceedings of IEEE VR 2002

40. Noel R. W. and Hunter C. M. (2000) Mapping the Physical World to Psychological Reality: Creating Synthetic Environments. In Proceedings of DIS 2000, ACM

41. Osawa N., Asai K. and Sugimoto Y. Y. (2000) Immersive Graph Navigation Using Direct manipulation and Gestures. In Proceedings of VRST 2000, ACM

42. Pactrick E., Cosgrove D., Slavkovic A., Rode J. A., Verratti T. and Chiselko G. (2000) Using a large Projection Screen as an alternative to Head-amounted Displays for Virtual Environments. In Proceedings of CHI 2000, ACM

43. Paelke V. (2000) Systematic Design of Interactive Illustration Techniques for User Guidance in Virtual Environments. In Proceedings of IEEE VR 2000

44. Passmore P. J., Nielsen C. F., Cosh W. J. and Darzi A. (2001) Effects of Viewing and Orientation on Path Following in a Medical Teleoperation Environment. In Proceedings of IEEE VR 2001

45. Pierce J. S. and Pausch R. (2002) Comparing Voodoo Dolls and HOMER: Exploring the Importance of Feedback in Virtual Environments. In Proceedings of CHI 2002, ACM

46. Pinho M. S., Bowman D. A. and Freitas C. M. D. S. (2002) Cooperative Object Manipulation in Immersive Virtual Environments: Framework and Techniques. In Proceedings of VRST 2002, Hong Kong, ACM

47. Poupyrev I., Weghorst S. and Fels S. (2000) Non-Isomorphic 3D Rotational Techniques. In Proceedings of CHI 2000, ACM

48. Reitmayr G. and Schmalstieg D. (2001) An Open Software Architecture for Virtual Reality Interaction. In Proceedings of VRST 2001, ACM

49. Ruddle R. A., Savage J. D. and Jones D. M. (2002) Implementing Flexible Rules of Interaction for Object Manipulation in Cluttered Virtual Environments. In proceedings of VRST 2002, Hong Kong, ACM

50. Sato Y., Saito M and Koike H. (2001) Real Time Input if 3D Pose and Gestures of a User’s Hand and Its Applications for HCI. In Proceedings of IEEE VR 2001

51. Schkolne S., Pruett M. and Schroder P. (2001) Surface Drawing: Creating Organic 3D Shapes with the Hand and Tangible Tools. In Proceedings of CHI 2001, ACM

52. Schroeder R., Steed A., Axelsson A., Heldal I., Abelin Å., Widestrom J., Nilsson A. and Slater M. (2001) Collaborating in networked immersive spaces: as good as being there together? Computers and Graphics (2001) Vol. 25, 781-788

53. Sheridan T. B. (2000) Interaction, Imagination and Immersion Some Research Notes. In Proceedings of VRST 2000, ACM

54. Song C. G., Kwak N. J. and Jeong D. H. (2000) Developing an efficient technique of Selection and Manipulation in Immersive V. E. In Proceedings of VRST 2000, ACM

55. Stoev S. L. and Schmalstieg D. (2002) Application and Taxonomy for Through-The-Lens Techniques. In proceedings of VRST 2002, Hong-Kong, ACM

56. Stuerzlinger W and Smith G. (2002) Efficient Manipulation of Object Groups in Virtual Environments. In Proceedings of IEEE VR 2002

57. Sun H. and Hujun B. (2002) Two-handed Assembly with Immersive Task Planning in Virtual Reality. Virtual Reality (2002) (6), 11-20, Springer-Verlag

58. Tanriverdi V and Jacobs R. J. K. (2001) VRID: A Design Model and Methodology for Developing Virtual Reality Interfaces. In Proceedings of VRST 2001, ACM

59. Tanriverdi V. and Jacob R. C. K. (2000) Interacting with eye movements in virtual environments. In Proceedings of CHI 2000, ACM

60. van Dam A., Laidlaw D. H. and Simpson R. M. (2002) Experiments in Immersive Virtual Reality for Scientific Visualization. Computers and Graphics (2002) Vol. 26, 535-555

61. Wang W., Lin Q., Ng J. M. and Low C. P. (2001) SmartCU3D: A collaborative Virtual Environment System with Behavior Based Interaction Management. In Proceedings of VRST 2001, ACM

62. Wang Y. and MacKenzie C. L. (2000) The Role of Conceptual Haptic and Visual Constraints on Object manipulation in Virtual Environments. In Proceedings of CHI 2000, ACM

63. Wesche G. and Seidet H. P. (2001) FreeDrawer – A Free-Form Sketching System on the Responsive Workbench. In Proceedings of VRST 2001, ACM

64. Wesche G and Droske M. (2000) Conceptual Free-Form Styling on the Responsive Workbench. In Proceedings of VRST 2000, ACM

65. Zeleznik R., LaViola Jr. J. J., Feliz D. A. and Keefe D. F. (2002) Pop Through Button Devices for VE Navigation and Interaction. In Proceedings of IEEE VR 2002

B.2 Classification of HCI Research Papers for Augmented Reality

Research Method

Research Purpose	Case studies	Field studies	Action research	Lab experiment	Survey research	Applied research	Basic research	Normative writings
Understand	24	7, 21		11, 25		32	29, 34	
Engineer		17, 18, 21, 30, 33	20			2, 3, 4, 5, 6, 8, 9, 10, 15, 26, 27, 28, 31, 37, 38, 39, 40, 41, 42, 44, 46	13	11, 12
Re-engineer						19, 23, 43, 47		
Evaluate		24, 26, 28, 44		5, 9, 11, 19, 23, 25, 27, 31, 40, 43	6			10, 32, 37
Describe		17				1, 15, 23, 42		14, 16, 22, 35, 36, 45

Table 2. Classification of HCI research papers for augmented reality

1. Balcisoy S., Kallmann M., Fua P. and Thalmann D. (2000) A Framework for rapid evaluation of prototypes with augmented reality. In Proceedings of VRST 2000, Seoul, Korea, ACM
2. Behringer R., Tam C., McGee J., Sundareswaran S. and Vassiliou M. A (2000) Wearable Augmented Reality Testbed for Navigation and Control, Built Solely with Commercial-off-the-shelf (COTS) Hardware, In Proceedings of IEEE AR 2000
3. Bell B., Höllerer T. and Feiner S. (2002) An Annotated Situation-Awareness Aid for Augmented Reality. In Proceedings of UIST 2002, ACM
4. Bell, B., Feiner S. and Höllerer. T. (2001) View Management for Virtual and Augmented Reality. In Proceedings of UIST 2001, ACM
5. Billinghurst M., Kato H., Kiyokawa K., Belcher D. and Poupyrev I. (2002) Experments with Face-To-Face Collaborative AR Interfaces. Virtual Reality (2002) (6), 107-121, Springer-Verlag
6. Billinghurst M., Kato H. and Poupyrev I. (2001) The MagicBook: a transitional AR interface. Computers and Graphics (2001) Vol. 25, 745-753
7. Browall C. and Lindquist K. (2000) Our little orchestra: the development of an interactive toy. In proceedings of DARE 2000, ACM
8. Cheok A. D., Weihua W., Yang X., Prince S., Wan F. S., Billinghurst M. and Kato H. (2002) Interactive Theatre Experience in Ebodied + Wearable Mixed Reality Space. In Proceedings of IEEE AR 2002
9. Dorfmuller-Ulhaas K. and Schmalstieg D. (2001) Finger Tracking for interaction in augmented environments. In Proceedings of IEEE AR 2001
10. Fiorentino M., de Amicis R., Monno G. and Stork A. (2002) Spacedesign: A Mixed Reality Workspace for Aesthetic Industrial Design. In Proceedings of IEEE AR 2002

11. Fjeld M., Schar S. G., Signorello D. and Krueger H (2002) Alternative Tools for Tangible Interaction: A Usability Evaluation. In Proceedings of IEEE AR 2002

12. Fjeld M. (2000) Evaluating navigation methods for an AR system. In proceedings of DARE 2000, ACM

13. Geiger C., Paelke V., Reimann C. and Rosenbach W. (2000) A Framework for the Structured Design of VR/AR Content. In Proceedings of VRST 2000, Seoul, Korea, ACM

14. Grasso A., Karsenty A and Susani M. (2000) Augmenting Paper to Enhance Community Information Sharing. In Proceedings of DARE 2000, ACM

15. Hua H., Gau C. and Rolland J. P. (2002) A Testbed for Precise registration, natural occlusion and interaction in an augmented environment using head-mounted projective display (HMPD). In Proceedings of IEEE VR 2002

16. Höllerer T., Feiner S., Hallaway D., Bell B., Lanzagorta M., Brown D., Julier S., Baillot Y. and Rosenblum L. (2001) User interface management techniques for collaborative mobile augmented reality. Computers and Graphics (2001) Vol. 25, 799-810

17. Ishii H., Underkoffler J., Chack D., Piper B., Ben-Joseph E., Yeung L. and Kanji Z. (2002) Augmented Urban Planning Workbanch: Overlaying Drawings, Physical Models and Digital Simulation. In Proceedings of IEEE AR 2002

18. Johansson M. and Pettersson M. (2000) Eyes on the road – augmenting traffic information. In proceedings of DARE 2000, ACM

19. Kiyokawa K., Kurata Y. and Ohno H. (2001) An optical see-through display for mutual occlusion with a real-time stereovision system. Computers and Graphics (2001) Vol. 25, 765-779

20. Klinker G., Dutoit A. H. and Bauer M. (2002) Fata Morgana – A Presentation System for Product Design. In Proceedings of IEEE AR 2002

21. Klinker G., Creighton O., Dutoit A. H., Kobylinski R., Vilsmeier C. and Brugge B. (2001) Augmented maintenance of powerplants: A prototyping case study of a mobile AR system. In Proceedings of IEEE AR 2001

22. Klinker G., Reicher T. and Brugge B. (2000) Distributed User Tracking Concepts for Augmented Reality Applications, In Proceedings of IEEE AR 2000

23. Koike H., Sato Y and Kobayashi Y. (2001) Integrating Paper and Digital Information on EnhancedDesk: A Method for Realtime Finger Tracking on an Augmented Desk System. ACM Transactions on Computer-Human Interaction, TOCHI Vol. 8(4), 307-322

24. Koleva B., Taylor I., Benford S., Fraser M., Greenhalgh C., Schnadelbach H., Lehn D., Heath C., Row-Farr J. and Adams M. (2001) Orchestrating a Mixed Reality Performance. In Proceedings of CHI 2001, ACM

25. Laramee R. S. (2002) Rivalry and Interface with a Head-Mounted Display. ACM Transactions on Computer-Human Interaction, TOCHI Vol. 9(3), 238-251

26. Lehikoinen J. and Suomela R. (2002) WalkMap: Developing an Augmented Reality Map Application for Wearable Computers. Virtual Reality (2002) (6), 33-44, Springer-Verlag

27. Lehikoinen J. and Suomela R. (2002) Accessing Context in wearable computers. Personal and Ubiquitous Computing (2002) (6), 64-74

28. Ljungstrand P., Redstrom J. and Holmquist L. E. (2000) Webstickers: Using Physical Tokens to Access, Manage and Share Bookmarks to the Web. In Proceedings of DARE 2000, ACM

29. MacIntyre B., Bolter J. D., Moreno E. and Hannigan B. (2001) Augmented Reality as a New Media Experience. In Proceedings of IEEE AR 2001

30. Mackay W. E., Pothier G., Letondal C, Boegh K. and Soerensen H. E. (2002) The Missing Link: Augmenting Biology Laboratoy Notebooks. In Proceedings of UIST 2002, ACM

31. Malik S., McDonald C. and Roth G. (2002) Hand Tracking for Interactive Pattern-based Augmented Reality. In Proceedings of IEEE AR 2002

32. McGarry B., Matthews B. and Brereton M. (2000) Reflections on a candidate design of the user-interface for a wireless vital-signs monitor. In Proceedings of DARE 2000, ACM

33. McGee D. R., Cohen P. R. and Wu L. (2000) Something from nothing: Augmenting a paper-based work practice via multimodal interaction. In Proceedings of DARE 2000, ACM

34. Nicholas-Graham T. C., Watts L. A. Calvary G., Coutaz J. Dubois E. (2000) A Dimension Space for the Design of Interactive Systems within their Physical Environments. In Proceedings of DIS 2000, ACM

35. Pederson T. (2001) Magic Touch: A Simple Object Location Tracking System Enabling the Development of Physical-Virtual Artefacts in Office Environmenta. Personal and Ubiquitous Computing (2001) (5), 54-57

36. Pederson T. (2000) Human Hands as Link between Physical and Virtual. In Proceedings of DARE 2000, ACM

37. Pyssysalo T., Repo T., Turunen T., Lankila T and Roning J. (2000) CyPhone – Bringing Augmented Reality to Rext Generation Mobile Phones. In Proceedings of DARE 2000, ACM

38. Regenbrecht H., Wagner M. T. and Baratoff G. (2002) MagicMeeting: A Collaborative Tangible Augmented Reality System. Virtual Reality (2002) (6), 151-166, Springer-Verlag

39. Regenbrecht H., Baratoff G. and Wagner M. (2001) A tangible AR desktop environment. Computers and Graphics (2001) Vol. 25, 755-763

40. Reitmayr G. and Schmalstieg D. (2001) Mobile Collaborative Augmented Reality. In Proceedings of IEEE AR 2001

41. Rekimoto J. and Ayatsuka Y. (2000) Cybercode: Designing Augmented Reality Environments with Visual Tags. In proceedings of DARE 2000, ACM

42. Robinson J. A. and Robertson C. (2001) The LivePaper System: augmenting paper on an enhanced tabletop. Computers and Graphics (2001) Vol. 25, 731-743

43. Schmalstieg D., Fuhrmann A. and Hesina G. (2000) Bridging Multiple User Interface Dimensions with Augmented Reality, In Proceedings of IEEE AR 2000

44. Schnadelbach H. Koelva B., Flintham M., Fraser M., Izadi S., Chandler P., Foster M., Benford S., Greenhalgh C. and Rodden T. (2002) The Augorscope: A Mixed Reality Interface for Outdoors. In Proceedings of CHI 2002, ACM

45. Stricker D., Frohlich T. and Soller-Eckert C. (2000) The Augmented Man. In Proceedings of IEEE AR 2000

46. Thomas B., Close B., Donoghue J., Squires J., De Bondi P. and Piekarski W. (2002) First Person Indoor/Outdoor Augmented Reality Application: ARQuake. Personal and Ubiquitous Computing (2002) (6), 75-86

47. Thomas B. H. and Piekarski W. (2002) Glove Based User Interaction Techniques for Augmented Reality in an Outdoor Environment. Virtual Reality (2002) (6), 167-180, Springer-Verlag

B.3 Classification of HCI Research Papers for Mobile Computers

Research Method

Research Purpose

	Case studies	Field studies	Action research	Lab experiment	Survey research	Applied research	Basic research	Normative writings
Understand	13, 14, 55	71, 72, 73, 107		97	17, 28, 57, 76	48	1, 24, 35	19, 23
Engineer	27, 53	3, 69, 75, 91, 100				2, 5, 6, 11, 15, 20, 21, 22, 26, 30, 31, 32, 37, 38, 40, 42, 43, 46, 50, 51, 52, 54, 56, 61, 62, 63, 64, 68, 74, 77, 78, 80, 81, 82, 84, 83, 86, 87, 89, 90, 92, 95, 96, 98, 99, 101, 103, 104		
Re-engineer	25					9, 10, 12, 34, 59, 67, 79, 86, 106, 108		
Evaluate		8, 11, 46, 67, 75, 83, 86, 91, 95		5, 6, 7, 8, 9, 10, 11, 15, 16, 21, 25, 33, 36, 45, 47, 49, 53, 54, 59, 60, 62, 64, 65, 70, 79, 81, 88, 96, 100, 103, 104, 105	29, 37, 68, 99			
Describe		44, 66, 102		44		39		4, 18, 41, 58, 93, 94

Table 3. Classification of HCI research papers for mobile computers

1. Abowd G.D. and Mynatt E.D. (2000) Charting Past, Present and Future Research in Ubiquitous Computing. ACM Transactions on Computer-Human Interaction, TOCHI, Vol. 7(1), 29-58
2. Beigl M. (2000) MemoClip: A Location-Based Remembrance Appliance. Personal and Ubiquitous Computing (2000) Vol. (4), 230-233, Springer-Verlag
3. Bertelsen O. and Nielsen C. (2000) Augmented Reality as a Design Tool for Mobile Devices. In proceedings of DIS 2000, ACM
4. Bieber G. and Giersich M. (2001) Personal mobile navigation systems – design considerations and experiences. Computers and Graphics (2001) Vol. 25:563-570
5. Bohnenberger T., Jameson A., Kruger A. and Butz A. (2002) Location-Aware Shopping Assistance: Evaluation of a Decision-Theoretic Approach. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag
6. Brachtl B., Slajs J. and Slavik P. (2001) PDA based navigation system for a 3D environment. Computers and Graphics (2001) Vol. 25:627-634

7. Brewster S. and Murray R. (2000) Presenting Dynamic Information on Mobile Computers. Personal and Ubiquitous Computing (2000) Vol. (4), 209-212, Springer-Verlag

8. Brewster S. (2002) Overcoming the Lack of Screen Space on Mobile Computers. Personal and Ubiquitous Computing (2002) Vol. (6), 188-205. Springer-Verlag

9. Buyukkokten O., Garcia-Molina H., Paepcke A. and Winograd T. (2000) Power Browser: Efficient Web Browsing for PDAs. In Proceedings of CHI2000, The Hague, Amsterdam, ACM

10. Buyukkokten O., Garcia-Molina H., Paepcke A. (2001) Accordion Summarization for End-Game Browsing on PDAs and Cellular Phones. In proceedings of CHI2001, Seattle, WA, USA, ACM

11. Cheverst K., Davies N., Mitchell K., Friday A. and Efstratiou C. (2000) Developing a Context-aware Electronic Tourist Guide: Some Issues and Experiences. In Proceedings of CHI2000, The Hague, Amsterdam, ACM

12. Cheverest K., Smith G., Mitchell K., Friday A. and Davies N. (2001) The role of shared context in supporting cooperation between city visitors. Computers and Graphics (2001) Vol. 25:555-562

13. Cheverst K., Mitchell K. and Davies N. (2001) Investigating Context-aware Information Push vs. Information Pull to Tourists. In Proceedings of MobileHCI 2001, Lille, France

14. Cheverst K., Davies N., Mitchell K. and Efstratiou C. (2001) Using Context as a Crystal Ball: rewards and Pitfalls. Personal and Ubiquitous Computing (2001) (5), 8-11, Springer-Verlag

15. Chincholle D., Goldstein M., Nyberg M. and Eriksson M. (2002) Lost or Found? A Usability Evaluation of a Mobile Navigation and Location-Based Service. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag

16. Chittaro L. and Cin P.D. (2001) Evaluating Interface Design Choices on WAP Phones: Single-choice List Selection and Navigation among Cards. In Proceedings of MobileHCI 2001, Lille, France

17. Colbert M. (2002) A Diary Study of Rendezvousing: Group Size, Time, Pressure and Connectivity. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag

18. Constas I. and Papadopoulos D. (2001) Interface-Me: Pursuing Sociability Through Personal Devices. Personal and Ubiquitous Computing (2001) (5), 195-200, Springer-Verlag

19. Coschurba P., Baumann J., Kubach U. and Leonhardi A. (2001) Metaphors and Context-Aware Information Access. Personal and Ubiquitous Computing (2001) (5), 16-19, Springer-Verlag

20. Danesh A., Inkpen K., Lau F., Shu K. and Booth K. (2001) Geney: Designing a Collaborative Activity for the Palm Handheld Computer. In proceedings of CHI2001, Seattle, WA, USA, ACM

21. De Bruijn O., Spence R. and Chong M.Y. (2001) RSVP Browser: Web Browsing on Small Screen Devices. In Proceedings of MobileHCI 2001, Lille, France

22. Dey A.K. (2001) Understanding and Using Context. Personal and Ubiquitous Computing (2001) Vol. 5:4-7, Springer-Verlag

23. Dix A., Rodden T., Davies N., Trevor J., Friday A. and Palfreyman K. (2000) Exploiting Space and Location as a Design Framework for Interactive Mobile Systems. ACM Transactions on Computer-Human Interaction, TOCHI, Vol. 7(3), 285-321

24. Dubois E., Gray P., and Nigay L. (2002) ASUR++: A Design Notation for Mobile Mixed Systems. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag

25. Ebling M.R., John B.E. and Satyanarayanan S. (2002) The Importance of Transludence in Mobile Computing Systems. ACM Transactions on Computer-Human Interaction, TOCHI, Vol. 9(1), 42-67

26. Fano A. (2001) What are a Location's File and Edit Menus? Personal and Ubiquitous Computing (2001) (5), 12-15, Springer-Verlag

27. Forlizzi J. and McCormack M. (2000) Case Study: User Research to Inform the Design and Development of Integrated Wearable Computers and Web-Based Services. In proceedings of DIS 2000, ACM

28. Fortunati L. (2001) The Mobile Phone: An Identity on the move. Personal and Ubiquitous Computing, (2001) (5), 85-98, Springer-Verlag

29. Frohlich D. and Murphy R. (2000) The Memory Box. Personal and Ubiquitous Computing (2000) (4), 238-240, Springer-Verlag
30. Geldof S. and Terken J. (2000) Talking Wearables Exploit Context. Personal and Ubiquitous Computing (2000) (5), 62-65, Springer-Verlag
31. Gelgon M. and Tilhou K. (2002) Automated Multimedia Diaries of Mobile Device Users Need Summarization. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag
32. Goldstein M., Oqvist G., Bayat-M M., Ljungstrand P. and Bjork S. (2001) Enhancing the Reading Experience: Using Adaptive and Sonified RSVP for Reading on Small Displays. In Proceedings of MobileHCI 2001, Lille, France
33. Goldstein M., Alsio G. and Werdenhoff J. (2002) The Media Equation Does Not Always Apply: People are not Polite Towards Small Computers. Personal and Ubiquitous Computing (2002) (6), 87-96, Springer-Verlag
34. Gonzalez-Castano F.J., Anido-Rifon L. and Costa-Montenegro E. (2002) A New Transcoding Technique for PDA Browsers, Based on Context Hierarchy. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag
35. Green N., Harper R.H.R. and Cooper G. (2001) Configuring the Mobile User: Sociological and Industry Views. Personal and Ubiquitous Computing (2001) (5), 146-156, Springer-Verlag
36. Haan G D. and Terken J. M. B. (2001) Agents and wearables – usability in the COMRIS system. Computers and Graphics (2001) Vol. 25:571-579
37. Hibino S. and Mockus A. (2002) handiMessenger: Awareness-Enhanced Universal Communication for Mobile Users. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag
38. Hinckley K., Pierce J., Sinclair M. and Horvitz E. (2000) Sensing Techniques for Mobile Interaction. In proceedings of UIST2000, San Diego, CA, USA, ACM
39. Hinckley K. and Horvitz E. (2001) Toward More Sensitive Mobile Phones. In proceedings of UIST2001, Orlando, Florida, USA, ACM
40. Holland S. and Morse D.R. (2001) AudioGPS: spatial audio in a minimal attention interface. In Proceedings of MobileHCI 2001, Lille, France
41. Holland S., Morse D. and Gedenryd H. (2002) Direct Combination: A New User Interaction Principle for Mobile and Ubiquitous HCI. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag
42. Huang E.M., Terry M., Mynatt E., Lyons K. and Chen A. (2002) Distributing Event Information by Simulating Word-of-Mouth Exchanges. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag
43. Huttenrauch H. and Norman M. (2001) PocketCERO – mobile interfaces for service robots. In Proceedings of MobileHCI 2001, Lille, France
44. Iacucci G., Kuutti K. and Ranta M. (2000) On the Move with a Magic Thing: Role Playing in Concept Design of Mobile Services and Devices. In proceedings of DIS 2000, ACM
45. Isokoski P. and Raisamo R. (2000) Device Independent Text Input: A Rationale and an Example. In Proceedings of AVI2000, Palermo, Italy, ACM
46. Izadi S., Fraser M., Benford S., Flintham M., Greenhalgh C., Rodden T. and Schnadelbach H. (2001) Citywide: supporting interactive digital experiences across physical space. In Proceedings of MobileHCI 2001, Lille, France
47. James C.L. and Reischel K. (2001) Text Input for Mobile Devices: Comparing Model Prediction to Actual Performance. In proceedings of CHI2001, Seattle, WA, USA, ACM
48. Jameson A. (2001) Modeling both the Context and the User. Personal and Ubiquitous Computing (2001) (5), 29-33, Springer-Verlag
49. Jones M., Buchanan G. and Thimbleby H. (2002) Sorting Out Searching on Small Screen Devices. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag
50. Kehr R. and Zeidler A. (2000) Look, Ma, My Homepage is Mobile! Personal and Ubiquitous Computing (2000) (4), 217-220, Springer-Verlag

51. Kohtake N., Rekimoto J. and Anzai Y. (2001) InfoPoint: A Device that Provides a Uniform User Interface to Allow Appliances to Work Together over a Network. Personal and Ubiquitous Computing (2001) (5), 264-274, Springer-Verlag

52. Laerhoven K.V. and Aidoo K. (2001) Teaching Context to Applications. Personal and Ubiquitous Computing (2001) (5), 46-49, Springer-Verlag

53. Lamming M., Eldridge M., Flynn M., Jones C. and Pendlebury D. (2000) Satchel: Providing Access to Any Document, Any Time, Anywhere. ACM Transactions on Computer-Human Interaction, TOCHI, Vol. 7(3), 322-352

54. Lehikoinen J. and Salminen I. (2002) An Empirical and Theoretical Evaluation of BinScroll: A Rapid Selection Technique for Alphanumeric Lists. Personal and Ubiquitous Computing (2002) (6), 141-150, Springer-Verlag

55. Licoppe C. and Heurtin J.P. (2001) Managing One's Availability to Telephone Communication Through Mobile Phones: A French Case Study of the Development of Dynamics of Mobile Phone Use. Personal and Ubiquitous Computing (2001) (5), 99-108, Springer-Verlag

56. Lin J., Laddaga R. and Naito H. (2002) Personal Location Agent for Communicating Entities (PLACE). In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag

57. Ling R. (2001) We Release Them Little by Little: Maturation and Gender Identity as Seen in the Use of Mobile Telephony. Personal and Ubiquitous Computing (2001) (5), 123-136, Springer-Verlag

58. Ljungstrand P. (2001) Context Awareness and Mobile Phones. Personal and Ubiquitous Computing (2001) (5), 58-61, Springer-Verlag

59. MacKenzie I.S., Kober H., Smith D., Jones T. and Skepner E. (2001) LetterWise: Prefix-Based Disambiguation for Mobile Text Input. In proceedings of UIST2001, Orlando, Florida, USA, ACM

60. MacKenzie S. (2002) KSPC (Keystrokes per Character) as a Characteristic of Text Entry Techniques. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag

61. Mantyjarvi J. and Seppanen T. (2002) Adapting Applications in Mobile Terminals Using Fuzzy Context Information. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag

62. Marsden G., Thimbleby H., Jones M. and Gillary P. (2002) Data Structures in the Design of Interfaces. Personal and Ubiquitous Computing (2002) (6), 132-140, Springer-Verlag

63. Matsushita N., Ayatsuka Y. and Rekimoto J. (2000) Dual Touch: A Two-Handed Interface for Pen-Based PDAs. In proceedings of UIST2000, San Diego, CA, USA, ACM

64. Mayol W.W., Tordoff B.J. and Murray D.W. (2002) Wearable Visual Robots. Personal and Ubiquitous Computing (2002) (6), 37-48, Springer-Verlag

65. Mizobuchi S., Mori K., Ren X. and Michiaki Y. (2002) An Empirical Study of the Minimum Required Size and the Minimum Number of Targets for Pen Input on the Small Display. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag

66. Mikkonen M., Vayrynen S., Ikonen V. and Heikkila O. (2002) User and Concept Studies as Tools in Developing Mobile Communication Services for the Elderly. Personal and Ubiquitous Computing (2002) (6) , 113-124, Springer-Verlag

67. Milewski A. and Smith T.M. (2000) Providing Presence Cues to Telephone Users. In Proceedings of CSCW2000, Philadelphia, PA, USA, ACM

68. Nakanishi Y., Tsuji T., Ohyama M. and Hakozaki K. (2000) Context Aware Messaging Service: A Dynamical Messaging Delivery using Location Information and Schedule Information. Personal and Ubiquitous Computing (2000) (4), 221-224, Springer-Verlag

69. Nigay L., Salembier P., Marchand T., Renevier P. and Pasqualetti L. (2000) Mobile and Collaborative Augmented Reality: A Scenario Based Design Approach. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag

70. Oquist G. and Goldstein M. (2002) Towards an Improved Readability on Mobile Devices: Evaluating Adaptive Rapid Serial Visual Presentation. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag

71. Palen L., Salzman M. and Youngs E. (2000) Going Wireless: Behavior & Practice of New Mobile Phone Users. In Proceedings of CSCW2000, Philadelphia, PA, USA, ACM

72. Palen L., Salzman M. and Youngs E. (2001) Discovery and integration of Mobile Communications in Everyday Life. Personal and Ubiquitous Computing (2001) (5), 109-122, Springer-Verlag

73. Palen L. and Salzman M. (2002) Beyond the Handset: Designing for Wireless Communication Devices. ACM Transactions on Computer-Human Interaction, TOCHI, Vol. 9(2), 125-151

74. Partridge K., Chatterjee S. and Want R. (2002) TiltType: Accelerometer-Supported Text Entry for Very Small Devices. In proceedings of UIST2002, Paris, France, ACM

75. Pascoe J., Ryan N. and Morse D. (2000) Using While Moving: HCI Issues in Fieldwork Environments. ACM Transactions on Computer-Human Interaction, TOCHI, Vol. 7(3), 417-437

76. Perry M., O'Hara K., Sellen A., Brown B. and Harper R. (2001) Dealing with Mobility: Understanding Access Anytime, Anywhere. ACM Transactions on Computer-Human Interaction, TOCHI, Vol. 8(4), 323-347

77. Petrelli D., Not E., Zancanaro M., Strapparava C. and Stock O. (2001) Modeling and Adapting to Context. Personal and Ubiquitous Computing (2001) (5), 20-24, Springer-Verlag

78. Pham T., Schneider G., Goose S. and Pizano A. (2001) Composite Device Computing Environment: A Framework for Situated Interaction Using Small Screen Devices. Personal and Ubiquitous Computing (2001) (5), 25-28, Springer-Verlag

79. Pirhonen A., Brewster S. and Holguin C. (2002) Gestural and Audio Methaphors as a Means of Control for Mobile Devices. In Proceedings of CHI2002, Minneapolis, Minnesota, USA, ACM

80. Pospischil G., Umlauft M. and Michlmayr E. (2002) Designing LoL@, a Mobile Tourist Guide for UMTS. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag

81. Poupyrev I., Maruyama S. and Rekimoto J. (2002) Ambient Touch: Designing Tactile Interfaces for Handheld Devices. In proceedings of UIST2002, Paris, France, ACM

82. Raghunath M.T., Narayanaswami C. (2002) User Interfaces for Application s on a Wrist Watch. Personal and Ubiquitous Computing (2002) (6), 17-30, Springer-Verlag

83. Rakkolainen I. And Vainio T. (2001) A 3D City Info for mobile users. Computers and Graphics (2001) Vol. 25:619-625

84. Randell C. and Muller H. (2000) The Shopping Jacket: Wearable Computing for the Consumer. Personal and Ubiquitous Computing (2000) (4), 241-244, Springer-Verlag

85. Randell C. and Muller H.L. (2002) The Well Mannered Wearable Computer. Personal and Ubiquitous Computing (2002) (6), 31-36, Springer-Verlag

86. Rantanen J., Impio J., Karinsalo T., Reho A., Tasanen M. and Vanhala J. (2002) Smart Clothing Prototype for the Artic Environment. Personal and Ubiquitous Computing (2002) (6), 3-16, Springer-Verlag

87. Rauschenbach U., Jeschke S. and Schumann H. (2001) General rectangular fisheye views for 2D graphics. Computers and Graphics (2001) Vol. 25:606-617

88. Ren X., Moriya S. (2000) Improved Selection Performance on Pen-Based Systems: A Study of Pen-Based Interaction for Selection tasks. ACM Transactions on Computer-Human Interaction, TOCHI, Vol. 7(3), 384-416

89. Rist T., Brandmeier P., Herzog G. and Andre E. (2000) Getting the Mobile Users in: Three Systems that Support Collaboration in an Environment with Heterogeneous Communication Devices. In Proceedings of AVI2000, Palermo, Italy, ACM

90. Rist T., and Brandmeier P. (2001) Customizing Graphics for Tiny Displays of Mobile Devices. In Proceedings of MobileHCI 2001, Lille, France

91. Ross D.A. and Blasch B.B. (2002) Development of a wearable Computer Orientation System. Personal and Ubiquitous Computing (2002) (6), 49-63, Springer-Verlag

92. Roth J. and Unger C. (2001) Using Handheld Devices in Synchronous Collaborative Scenarios. Personal and Ubiquitous Computing (2001) (5), 243-252, Springer-Verlag

93. Roth J. (2001) Patterns of Mobile Interaction. In Proceedings of MobileHCI 2001, Lille, France

94. Ruuska-Kalliokulja S., Schneider-Hufschmidt M., Vaananen-Vainio-Mattila K. and Von Niman B. (2001) Shaping the future of Mobile Devices – Results of the CHI2000 Workshop on Future Mobile Device User Interfaces. In Proceedings of MobileHCI 2001, Lille, France

95. Sawhney N. and Schmandt C. (2000) Nomadic Radio: Speech and Audio Interaction for Contextual Messaging in Nomadic Environments". ACM Transactions on Computer-Human Interaction, TOCHI, Vol. 7(3), 353-383

96. Sazawal V., Want R. and Boriello G. (2002) The Unigesture Approach: One-Handed Text Entry for Small Devices. In Proceedings of MobileHCI 2002, Pisa, Italy, Springer-Verlag

97. Schenkman B.N. (2002) Perceived Similarities and Preferences for Consumer Electronics Products. Personal and Ubiquitous Computing (2002) (6), 125-131, Springer-Verlag

98. Schmidt A., Takaluoma A. and Mantyjarvi K. (2000) Context-Aware Telephony Over WAP. Personal and Ubiquitous Computing (2000) (4), 225-229, Springer-Verlag

99. Schmidt A., Stuhr T. and Gellersen H. (2001) Context-Phonebook – Extending Mobile Phone Applications with Context. In Proceedings of MobileHCI 2001, Lille, France

100. Sharples M., Corlett D. and Westmancott O. (2002) The Design and Implementation of a Mobile Learning Resource. Personal and Ubiquitous Computing (2002) (6), 220-234, Springer-Verlag

101. Silfverberg M., Mackanzie I.S. and Korhonen P. (2000) Predicting Text Entry Speed on Mobile Phones. In Proceedings of CHI2000, The Hague, Amsterdam, ACM

102. Strom G. (2001) Mobile Devices as Props in Daily Role Playing. In Proceedings of MobileHCI 2001, Lille, France

103. Swindells C., Inkpen K.M., Dill J.C. and Tory M. (2002) That one There! Pointing to establish device identity. In proceedings of UIST2002, Paris, France, ACM

104. Tang J., Yankelovich N., Begole B., Van Kleek M., Li F. and Bhalodia J. (2001) ConNexus to Awarenex: Extending awareness to mobile users. In proceedings of CHI2001, Seattle, WA, USA, ACM

105. Thomas B., Grimmer K., Zucco J. and Milanese S. (2002) Where Does the Mouse Go? An Investigation into the Placement of a Body-Attached TouchPad Mouse for Wearable Computers. Personal and Ubiquitous Computing (2002) (6), 97-112, Springer-Verlag

106. Trevor J., Hilbert D.M., Schilit B.N. and Koh T.K. (2001) From Desktop to Phonetop: A UI For Web Interaction On Very Small Devices. In proceedings of UIST2001, Orlando, Florida, USA, ACM

107. Weilenmann A. (2001) Negotiating Use: Making Sense of Mobile Technology. Personal and Ubiquitous Computing (2001) (5), 137-145, Springer-Verlag

108. Wobbrock J.O., Forlizzi J., Hudson S.E. and Myers B.A. (2002) WebThumb: Interaction Techniques for Small-Screen Browsers. In proceedings of UIST2002, Paris, France, ACM