

PenTable: Augmenting a PC Workplace with a Digital Tabletop

DANIEL LEITHINGER

DIPLOMARBEIT

eingereicht am
Fachhochschul-Masterstudiengang

DIGITALE MEDIEN

in Hagenberg

im August 2007

© Copyright 2007 Daniel Leithinger

Alle Rechte vorbehalten

Erklärung

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und die aus anderen Quellen entnommenen Stellen als solche gekennzeichnet habe.

Hagenberg, am 9. September 2007

Daniel Leithinger

Contents

Erklärung	iii
Acknowledgements	vii
Kurzfassung	viii
Abstract	ix
1 Introduction	1
1.1 Short History of PC Hardware Interfaces	1
1.2 The Graphical User Interface	3
1.3 The Gap Between the Physical and Digital World	5
1.4 Ubiquitous Computing	5
1.5 Digital Tabletops	6
1.6 Proposed Contribution	7
1.7 Thesis Outline	8
2 Background	9
2.1 Tabletop Input Devices	9
2.1.1 Anoto Tracking	11
2.2 Tabletop Displays	13
2.3 Tabletop GUI Metaphors	14
2.3.1 GUI for Setups Supporting Group Collaboration	14
2.3.2 Specific Challenges of Single User Setups	14
3 Related Work	15
3.1 Digital Tabletop Workplaces	15
3.1.1 Videodesk	15
3.1.2 DigitalDesk and EnhancedDesk	15
3.1.3 I/O Bulb and Sensetable	16
3.1.4 Roomware	16
3.1.5 BlueSpace	16
3.1.6 Augmented Surfaces	16
3.1.7 DiamondTouch	17

3.1.8	Shared Design Space	17
3.1.9	Microsoft Surface	17
3.1.10	Interactive environment-aware display bubbles	17
3.1.11	Visual Interaction Platform (VIP)	18
3.2	Peripheral and Ambient Displays	18
3.3	Direct Pen Input	19
3.4	Analysis	20
4	PenTable	22
4.1	Hardware Interfaces	22
4.2	Functionality	22
4.3	Features	24
5	Tabletop GUI	26
5.1	sWidgets Instead of Windows and Icons	26
5.1.1	Creating sWidgets	27
5.1.2	Modifying sWidgets	28
5.2	pMenus Instead of Menus	28
5.2.1	User Interaction	30
5.2.2	pMenu Metaphors	30
6	Implementation	35
6.1	Hardware Setup	35
6.1.1	Tabletop Display	35
6.1.2	Pen Input	35
6.2	Software Implementation	37
7	Example Applications	38
7.1	Note Taking	38
7.2	World Clock	39
7.3	Media Player Controls	40
7.4	Reminder	40
7.5	Virtual Graphics Tablet	41
8	Evaluation	43
8.1	sWidget Informal User Study	43
8.2	pMenu User Study	45
8.2.1	Introduction	45
8.2.2	Hypothesis	45
8.2.3	Experiment	45
8.2.4	Results	47
8.2.5	Discussion	52
8.3	Informal Evaluation of the PenTable Setup	54
9	Conclusion and Future Work	56

<i>CONTENTS</i>	vi
A Content of the CD-ROM	58
A.1 Thesis	58
A.2 Additonal Ressources	58
B pMenu User Study Materials	59
B.1 Consent Form	59
B.2 Post-Study Questionnaire	60
Bibliography	63

Acknowledgements

I would like to thank my family for their tremendous support throughout the last years. This thesis is possible due to the support of my supervisor Michael Haller and the helpful comments of my colleagues in Hagenberg.

Kurzfassung

Interaktive Tischflächen werden zunehmend eingesetzt, um die Schnittstellen zwischen Mensch und Computer natürlicher zu gestalten. Doch während aktuelle Systeme mit interaktiven Tischflächen erfolgreich für spezielle Anwendungsfälle wie grafische Gestaltung und Mehrbenutzer-Kooperation eingesetzt werden, ist ihre Verwendung für Mehrzweck-Arbeitsplätze bisher wenig erforscht. Für das *PenTable* System wird eine interaktive Tischfläche mit herkömmlichen PC Schnittstellen kombiniert, um einen solchen Arbeitsplatz zu schaffen. Die Tischfläche wird dabei als peripherer Bildschirm und als stiftbasiertes Eingabegerät verwendet.

Diese Diplomarbeit beschreibt Design und Anwendungen von PenTable und geht im Detail auf die grafische Benutzerschnittstelle (GUI) auf der Tischfläche ein. Diese besteht aus *sWidgets* und *pMenus*, welche für die spezifischen Anforderungen von interaktiven Tischen mit Stifteingabe entwickelt wurden. Darüber hinaus werden die Ergebnisse einer Benutzerstudie präsentiert, in der *pMenus* mit herkömmlichen Kontextmenüs verglichen werden.

Abstract

Digital tabletops are often utilized to create natural interfaces for human computer interaction. Previous research has introduced intuitive metaphors for specific interaction scenarios like sketching and multi-user collaboration. However, only few projects have investigated the challenges and possibilities of creating a multi-purpose workplace with a digital tabletop. The *PenTable* system combines such a tabletop with standard PC interfaces like keyboard, mouse and LCD display. In the setup, the digital tabletop is utilized as a peripheral display and direct pen input device.

This thesis describes design and applications of PenTable and provides a detailed description a special user-drawn graphical user interface (GUI) on the tabletop, which consists of *sWidgets* and *pMenus*. These elements have been custom designed for the specific requirements of direct input tabletop displays. The thesis also presents the results of a user-study which compares pMenus with standard pop-up menus.

Chapter 1

Introduction

In 1981, the *Xerox 8010 Star* was released, which is considered to be the first commercially available modern personal computer (PC)¹ [32]. Figure 1.1 depicts the hardware interfaces of a Xerox Star: a keyboard, a mouse and a monitor with a graphical user interface (GUI). These interfaces are very similar to those of a modern desktop computer, which can be seen in Figure 1.2. While computers have vastly improved over the last centuries and become a crucial part of almost every office workplace [43], their interfaces have mostly remained unchanged. This thesis provides an overview of alternative interface approaches to improve human computer interaction (HCI). It also describes the design and implementation of *PenTable*, a PC workplace augmented with a digital tabletop.

1.1 Short History of PC Hardware Interfaces

The first program with a GUI, *Sketchpad*, was developed by Iven Sutherland in 1963 [65]. It was operated with a light pen directly on the computer screen. The light pen was at that time a common hardware pointing device also used (in a gun form factor) by the operators of the *SAGE* console [30]. The dominance of pen-shaped pointing devices changed, however, after the introduction of the *oNLine System (NLS)* [14]. When Douglas Engelbart's research group at *SRI* investigated the best display selection means for the NLS in 1967 [15], they compared a number of input devices with Engelbart's own invention, the *mouse*. They observed that although inexperienced users performed fastest with a light pen followed by a custom *knee control device*, experienced users preferred the mouse. Participants also complained about fatigue effects, when working with the light pen over extended periods of time. Because of these findings, the mouse was chosen as the selection device

¹While the abbreviation PC is often used exclusively for "IBM PC compatible" computers running Microsoft Windows, it also applies to other personal computers with Mac OS or Linux.



Figure 1.1: The hardware and software interfaces of a Xerox 8010 Star, released in 1981. (Courtesy of Xerox Corporation, thanks to Dave Curbow)



Figure 1.2: The hardware and software interfaces of an Apple iMac, released in 2007. (Courtesy of Apple)

for the NLS [45, p. 29]. Subsequently, it was also used for the Xerox Alto and Star computers [45, p. 43], and later the Apple Lisa and Macintosh [45, p. 107]. Today, the mouse is the default pointing device for desktop PCs [45, p. 115]. While its form factor, sensor technology and connectivity have been constantly refined, the way of interacting with it has remained unchanged. However, alternative pointing interfaces are commonly used in the graphics industry and for mobile computing.

Graphics tablets like the *RAND tablet* [10] have been used as input devices prior to the invention of the mouse and are still widely utilized in the graphics industry. Their main advantage is a haptic drawing experience similar to that of pen and paper. Also, the pen tip pressure information is often used as an additional input parameter for graphics programs. Most commercially available graphics tablets are used as an indirect input device, but solutions like the Wacom Cintiq² integrate the pen input directly into the screen to provide direct input. The same pen input technology is also used for most tablet PCs. While *slate style* tablet PCs rely solely on direct input on the screen, *convertible* tablet PCs have the form factor of a regular notebook computer, which can be converted into a slate. Speech recognition is part of the Microsoft Tablet PC interfaces, but most of the currently available software does not utilize it.

Pointing devices other than mouse and pen are used for mobile computing, gaming and media center controls. They include pointing sticks, touchpads, joysticks and rotation sensors. Such devices are often utilized in environments lacking appropriate space for mouse input.

1.2 The Graphical User Interface

When the *Xerox Alto* was established in the 1970's at Xerox PARC, it was the first PC to feature a GUI consisting of windows, icons, menus and a pointing device (WIMP). The GUI of modern operating systems like OS X and Windows Vista still consists of these elements. HCI experts have criticized the limitations of the WIMP interface [20], and as a result alternative approaches have been developed [9]. But none of these alternative solutions have replaced WIMP interaction up to now.

Metaphors are featured in every WIMP GUI [20, p. 72]. The desktop metaphor, for instance, represents an office workplace with a tabletop, a filing cabinet and a trash can. Another example can be seen in Figure 1.3, where physical image manipulation tools like scissors, pens and erasers, are featured as metaphors in Adobe Photoshop. These tools are more powerful than their real world counterparts, but less intuitive to use, as a comparison with Figure 1.4 highlights. The capabilities of computer programs improve

²<http://www.wacom.com/cintiq>

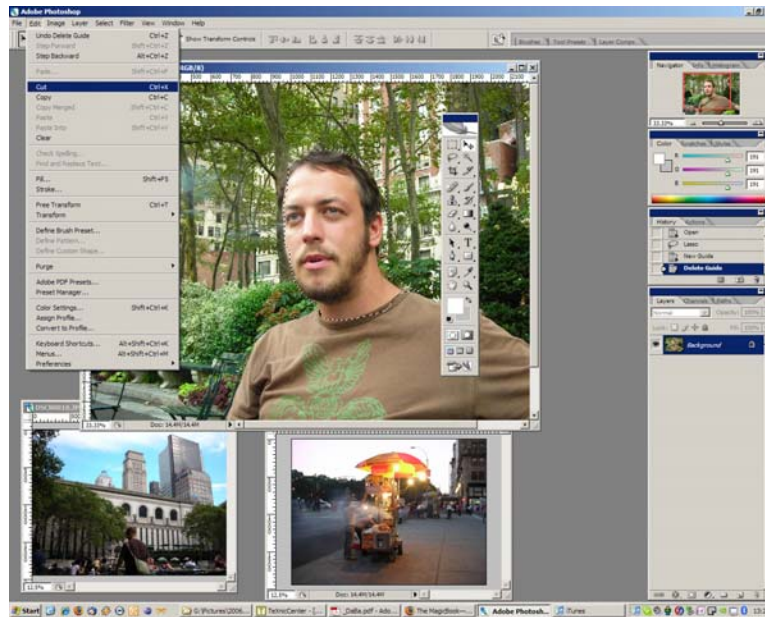


Figure 1.3: Cutting out part of an image in Adobe Photoshop CS2. The desired part is selected with the lasso tool and a cut command is issued.



Figure 1.4: Cutting out part of an image with a scissor. While tools like scissors and pens are less powerful than their digital counterparts, they are often more intuitive to use.

with every new release, while at the same their interfaces seem inadequate to capture the richness of interacting with physical content.

1.3 The Gap Between the Physical and Digital World

As the functionality of physical tools is often copied and even surpassed by computer programs, many physical items seem rendered unnecessary in modern offices. Paper books and documents, Post-It notes, calculators or telephones could be replaced with computer programs to reduce costs and save space. Therefore, efforts have been made to eliminate these physical items through various means of office automation [22].

An example is the integration of e-mail into office communication, which was supposed to reduce printing costs. However, according to Sellen and Harper [59, p. 13], paper consumption has increased by an average of 40 percent in organizations switching to e-mail. Although digital documents can be organized, stored and searched with less effort than paper documents, many people seem to prefer the physical attributes of paper. It does not require batteries, is lightweight and easy to annotate. So while the content is the same across different mediums (like PC and paper), the physical properties of each medium determine how comfortable users feel interacting with that content.

Since the physical properties of a PC are determined by its interfaces, a large number of research projects develop alternatives for a more natural computer interaction. Two very different approaches to create such computer interfaces are *Virtual Reality* (VR) and *Ubiquitous Computing*. Figure 1.5 (a) depicts a user wearing typical VR interfaces, which are designed to immerse him into a simulated environment. This sense of immersion is created by distancing the user as much as possible from the surrounding physical environment. Examples of ubiquitous computing interfaces are shown in Figure 1.5 (b), where a digital tabletop and an interactive whiteboard are integrated into a meeting room to a point of becoming an indistinguishable part of the users environment.

1.4 Ubiquitous Computing

“The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it.” (*Mark Weiser* [69])

When Mark Weiser coined the term *ubiquitous computing* in the early 1990s, he described a vision of computers being seamlessly integrated into the users physical environment. He criticized the traditional desktop PC interfaces as

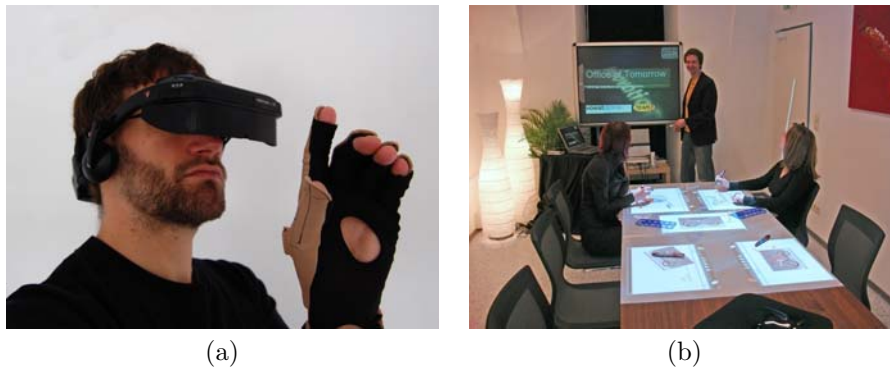


Figure 1.5: Typical Virtual Reality (VR) interfaces: A headmounted display and a data glove (a). Examples of ubiquitous computing interfaces: An augmented meeting room table and an interactive whiteboard (b).

demanding too much of the users' attention. Instead of a single PC per user, he promoted the idea of an environment filled with small, specialized and interconnected computers. Each of these would have an interface specifically designed for its designated purpose, allowing the user to focus on the task instead of the tool.

Weiser and his colleagues at Xerox PARC built a number of prototypes they called *tabs*, *pads* and *boards*. *Tabs* are small computers (the size of Post-It notes), which are used as personal ID tags, calculators and organizers. Applications are displayed on *Pads* (the size of a book) instead of program windows. *Boards* are digital whiteboards, which allow users to sketch and present ideas. Ubiquitous computing is not defined through any of these individual devices, but how they are combined with each other and the way they are designed to enable effortless user interaction. The challenges of creating such an environment are described in Weiser's paper "The computer for the 21st century" [69].

Ubiquitous computing has influenced a number of commercial products like PDAs (personal digital assistants), tablet PCs and digital whiteboards. Weiser's vision also had an impact on numerous research projects, which introduce new interfaces for PC interaction by augmenting physical objects or surfaces. Digital tabletops are an example for such interfaces.

1.5 Digital Tabletops

Interacting with an augmented tabletop as a computer interface is not a new idea. It is already proposed in the visionary article "As We May Think" by Vannevar Bush in 1945 [7], which was highly influential to following computer interface designs [62]. Bush describes an office desk with monitors built into the tabletop that allow the user to access content on microfilm.

While Bush’s vision has eventually led to the development of the current desktop PC [45, p. 30], tabletops have not been utilized as an interactive display until Wellner’s introduction of DigitalDesk in 1991 [70]. Since then, a large number of research projects have focused on the development of digital tabletop setups that provide either one of the following functionalities:

1. *Shared group workplace* to mediate interaction and collaboration between co-located users. Most of the current tabletop research focuses on this functionality, since standard PC interfaces are perceived insufficient to effectively support group interaction.
2. *Digital workbench* with a touch or pen-based interface for single users. Several projects have proposed different forms of digital single user workbenches. Most of them have been developed for a single specific task like sketching and therefore promoted special metaphors to support that task. While these workbenches provide very natural interfaces for such specific tasks, they often lack the versatility of a standard PC. For this reason, none of the existing research projects integrate a digital tabletop into a multi-purpose setup.
3. *Peripheral display* to provide ambient information. Peripheral displays are an emerging area, but only few projects have investigated the characteristics of utilizing tabletop displays for this functionality.

1.6 Proposed Contribution

We propose *PenTable*, an office workplace combining a desktop PC with an augmented tabletop. The PenTable system derives from previous research, as it uses a digital tabletop as an alternative computer interface. However, unlike most previous research projects, the PenTable setup explicitly combines the digital tabletop with standard PC interfaces like keyboard, mouse and monitor to create a multi-purpose workplace. The digital tabletop enhances working with the PC by acting as a pen-based input device and a peripheral display. This thesis describes the design considerations of the system and the challenges of implementing it. It also includes a detailed description of two custom developed GUI metaphors for tabletop displays, user-drawn path menus (*pMenus*) and sketched widgets (*sWidgets*), and presents the results of a study comparing pMenus with standard pop-up menus. The pMenu metaphors and the results of the user-study are also going to be published under the title “Improving Menu Interaction for Cluttered Tabletop Setups with User-Drawn Path Menus” in the Proceedings of the “2nd IEEE Tabletop 2007 Workshop”.

1.7 Thesis Outline

Chapter 2 covers the background and chapter 3 provides an overview of the related work. The design considerations that went into the PenTable system are explained in chapter 4. Details about the GUI are provided in chapter 5 and chapter 6 describes the implementation. Some example applications are explained in chapter 7. Chapter 8 describes the system evaluation and chapter 9 concludes the thesis and provides an outlook of future work.

Chapter 2

Background

As stated by Stacey Scott and Sheelagh Carpendale in 2006 [57], both the hardware and software aspects of digital tabletop setups continue to evolve and branch, thus providing a wide variety of interaction possibilities and many opportunities to explore alternative design approaches. For this reason, most of the current tabletop research projects either focus on solving hardware challenges like providing adequate input devices and displays, or the software challenges of developing special GUI metaphors. Both of these challenges are described in more detail in the following sections.

2.1 Tabletop Input Devices

The main criteria for tabletop input devices are scalability, precision, price, robustness and ease of use. The following list describes a selection of hardware solutions currently used for digital tabletop setups.

1. **Indirect input** is used in setups like *Augmented Surfaces* [54] (laptop pointing sticks) and *Coeno* [23] (cordless mouse and presenter). Findings by Forlines et al. [19] point out single user scenarios where mouse input for tabletop interaction might be more appropriate than direct input devices like touchscreens and pens.
2. **Voice input** is utilized to issue input commands in combination with other input devices like the *DiamondTouch* surface [67] and the *InteracTable* [34].
3. **External optical tracking** utilizes one or more cameras to track user input on the table surface. The cameras usually track physical objects [68], hands [36], fingers [70], pens [70] or laser pointer dots [8]. To retrieve additional information like object identification and rotation, special markers are sometimes utilized [33, 54].

Examples of tracking with ceiling-mounted cameras include *Urp* [68] and the *DigitalDesk* [70] setup. Alternatively, multiple cameras can be mounted into a frame on top of the display, as in the *SMARTBoard DViT* [28] setup, or underneath the table surface [25, 33]. While a rear-mounted camera requires a transparent table surface, it solves the problem of objects obscuring the camera. Another common problem of camera-based systems is the camera resolution, which limits the tracking resolution. In addition, tracking systems based on external camera tracking are easily to scale because of the cameras limited field of view.

Instead of cameras, rows of infrared light-emitting diodes (LED) and photoreceptors can be built into the borders of a tabletop display to determine user input. This tracking technology is utilized for the *Entertable* setup [29], which is able detect up to 40 points of contact on a 32-inch tabletop display simultaneously.

4. **Electromagnetic tracking**, commonly used in graphics tablets and tablet PCs, can also be built into table surfaces. A digitizing screen tracks either objects containing a coil or a battery on its surface. This technology is also able to determine the position objects hovering over the surface without touching it, but is limited to a number of input devices and to a certain size. It is utilized in the *Sensetable* setup for user input via physical objects [50].
5. **Analog resistive tracking** is often used for single point interaction touchscreens of various sizes. It consists of two electrically conductive and resistive layers, which are separated from each other with air in between them. When pressure is applied to the layers, they touch each other and the point of contact can be determined. An analog resistive *Matisse PP250* touchscreen [63] is integrated into the *InteracTable* [66], a meeting table with a 50-inch tabletop plasma display. This setup is commercially available by *Wilkhahn*¹.
6. **Weight tracking** is applied at the *WeightTable* developed by Schmidt et al. [56]. Touch input is tracked by measuring the pressure applied to a table surface with weight sensors built into the corners of the tabletop.
7. **Sonic tracking** is used by commercially available devices like *Mimio*² and *eBeam*³. The input pen sends an ultrasonic signal, which is captured by the microphones of a receiver. The pen position is then determined via triangulation. This tracking technology was developed for

¹<http://www.wilkhahn.de>

²<http://www.mimio.com>

³<http://www.e-beam.com>

electronic whiteboards, but can be applied to horizontal surfaces [60]. A problem is the decreasing tracking quality when occlusions between pen and receiver occur, for instance because someone leans on the tabletop.

Another version of sonic tracking uses *surface acoustic waves (SAW)*, which are generated by vibrating a glass surface on top of the display with transducers. When these waves are interrupted because someone touches the surface, reflectors determine the point of contact. Advantages of this technology are a high tracking resolution and high durability. SAW tracking can only determine single touch input and only works with soft objects (e.g. gloves or fingers) touching the surface. An SAW touchscreen is integrated into the *Tangible TableTop (TTT)* interface [37].

8. **Magnetic tracking** is applied for a 6 degree-of-freedom tracking solution as used in the *TractorBeam* setup [49]. This technology has the advantage of not being restricted to tracking on 2D surfaces, but is limited to a certain radius, expensive and easily interferes with other electronic devices.
9. **Capacitive tracking** with very small amounts of current conducted through the users bodies is applied in the *DiamondTouch* setup [11]. It provides precise and robust multi touch, but requires users to stay in contact with special conductive pads.
10. **Anoto tracking** utilizes a little camera built into the tip of a pen to determine its position on a unique dot pattern. The pen then sends this position to a connected PC. Since *PenTable* uses this technology as a tracking solution, it is described in more detail in the following section.

2.1.1 Anoto Tracking

The technology used to track pen input on the PenTable tabletop surface was developed by Anoto⁴. Figure 2.1 depicts a schematic view of the electronic components of a digital pen. An infrared diode and camera are built into the tip of the pen. The camera recognizes a dot pattern printed on paper, as shown in figure 2.2. Each dot of the pattern is aligned to a grid, but has a small displacement to this grid, so it can be positioned in 4 different possibilities. The pen camera captures 6×6 dots at a time, which therefore allows 4^{36} different combinations. Since the dots are 0,3 mm apart from each other, an area of approximately 60 million km² can be covered with unique pattern. The pen tip also contains a pressure sensor, so the pen recognizes

⁴<http://www.anoto.com>

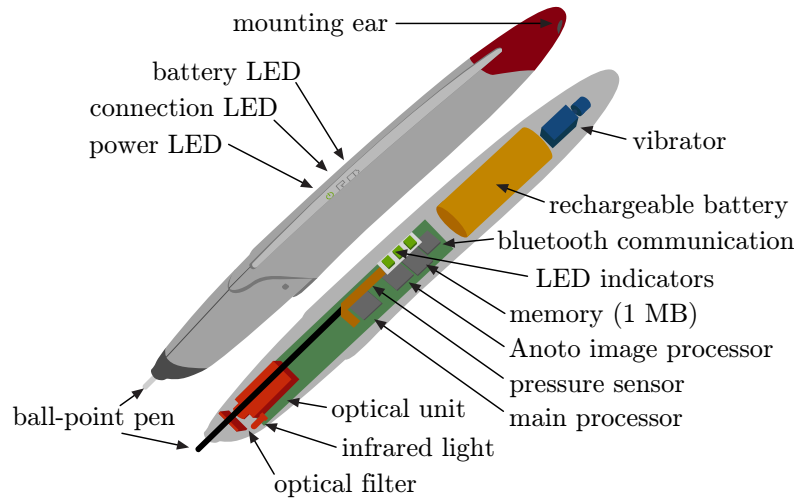


Figure 2.1: Schematic view of the components of an Anoto digital pen. (Courtesy of Jakob Leitner)

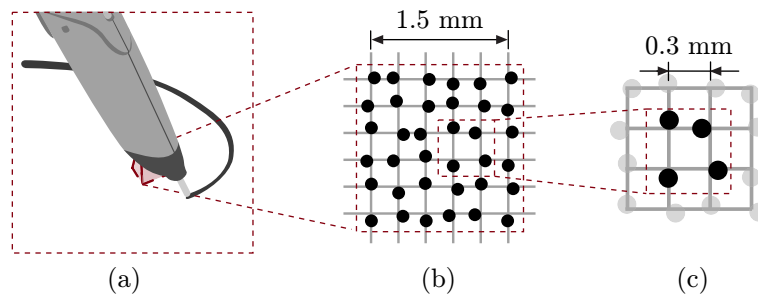


Figure 2.2: The pen camera captures 6×6 dots to determine its position on the Anoto pattern. (Courtesy of Jakob Leitner)

when it is touching the surface. The position, pen tip pressure and current time stamp are either stored in the pen memory for later synchronization or streamed to a connected PC via Bluetooth. Pen manufacturers include Logitech, Maxell and Nokia.

Advantages: The main advantage of using Anoto technology compared to other tabletop input devices is the high tracking precision on large surfaces. Since all of the technical components are built into the pen, multiple input devices do not disturb each other. The pen tracking is not affected by physical objects on the tabletop surface, as long as they do not occlude the pattern in the camera's field of view. The tracking surface can easily be extended by adding more pattern printouts. Although the tracking is

limited to 2D, it is not constrained to a single surface and can be applied to 3D objects by covering them with Anoto pattern, as demonstrated by Song et al. [64].

Disadvantages: A disadvantage of this technology is that the pen tracking only works on Anoto pattern, which needs to be protected by a scratch resistant cover. If an object like a piece of paper obscures the pattern, the tracking fails. Since the camera needs to capture the dot pattern on the paper, very fast pen movements or a poor pattern printout can result in a decreased tracking quality. In addition, the pen position is only determined when the pen tip touches the surface, therefore pen operations requiring hovering in mid-air are not supported. If the signal strength of the Bluetooth connection to the PC is weak, a lag between pen input and receiving the pen data can occur. Since the pen is battery-operated, it needs to be recharged after approximately 4 hours of continuous use. At the time of writing, royalties have to be paid to Anoto to get a license for printing and redistributing the tracking pattern.

2.2 Tabletop Displays

The display is a critical factor of any digital tabletop setup, as it needs to be robust and of a sufficient size and resolution. Other requirements like shape or portability of the setup also influence the display choice and therefore no standard solution exists at the moment.

1. **LCD or Plasma displays:** Large LCD or Plasma screens are built into horizontal surfaces as in setups like the *iLounge* [5], *Roomware* [53] or the *Philips Entertaible* [29]. A drawback of such a construction is the size and shape limitations of the displays. Therefore, projects like the *DAP environment* [46] use an array of combined LCD displays to provide a higher resolution and scalability at the cost of display disturbances caused by the screen bezels.
2. **Projectors:** Front or rear projection is the most commonly used technology for tabletop displays. Top projection is the only tabletop display technology, which does not require a custom table construction. A drawback of a top-projection setup is that the projector has to be mounted on the ceiling, resulting in a limited portability. When the table is moved, resulting offsets between input device and display can require a recalibration of the tracking solution. The users' body parts cast a shadow when interacting with the tabletop, which can be disturbing. Wilson compares the characteristics of the various forms of tabletop projection setups and proposes a portable solution with *PlayAnywhere* [71].

3. **Electronic Ink Displays:** *E Ink* displays consist of millions of micro-capsules with positively charged white particles and negatively charged black particles. When applying charge to a capsule, the particles inside move and its surface color changes. Currently commercially available *E Ink* displays only support grayscale images and are limited to 10-inch diameter, but prototypes supporting color and larger screen sizes up to 40-inch have been demonstrated [12]. The low power consumption, flexibility, robustness and good readability of *E Ink* are features that could greatly improve the usability of future tabletop setups.

2.3 Tabletop GUI Metaphors

2.3.1 GUI for Setups Supporting Group Collaboration

When multiple users sit around the tabletop and view it from different angles, software problems arise that are not addressed by current standard software. Many WIMP GUI elements like text and menus can not be used with the same convenience when viewed from different angles. Some common interface metaphors like 3D edges of pressed and unpressed buttons even lose their meaning when viewed upside down. In addition, operating systems like OS X or Windows XP can only handle a single pointing device and therefore do not support simultaneous input from multiple users.

Because of these constraints, almost all group collaboration setups feature a custom GUI. They are not designed for and usually not compatible with existing software like standard office automation and creativity applications.

2.3.2 Specific Challenges of Single User Setups

Single user setups do not face software interface problems like multiple user input or different viewing angles. Commercial software designed for desktop computers can be executed on such systems [61]. But since the GUI of WIMP applications has not been designed for the tabletop interfaces, it is questionable if productivity or user satisfaction increases on the tabletop setup. For some tasks like sketching, pen input directly on a horizontal display is very useful, other tasks are easier to accomplish with standard PC interfaces [13]. When designing a multi-purpose setup, a major challenge is to provide an interface for tasks, which benefit from the tabletop surface, while not sacrificing the usability of applications that do not. For such applications, the digital tabletop could still enhance the work experience as a peripheral display.

Chapter 3

Related Work

3.1 Digital Tabletop Workplaces

3.1.1 Videodesk

Videodesk was developed by Myron Krueger in 1985 as a variation of his *Videoplace* installation [36]. The system consists of a desk with a top mounted camera to track user hand movements over the desk surface in 2D. These movements are interpreted as input for a drawing application shown on a separate screen. Since it does not feature a tabletop display, the Videodesk setup is not a “digital tabletop” workplace in the sense of the following projects.

3.1.2 DigitalDesk and EnhancedDesk

DigitalDesk was introduced by Pierre Wellner in 1991 [70]. Wellner’s primary objective was to blur the boundaries between desktop computers and paper documents used in offices. He considered the process of scanning and printing documents too cumbersome and proposed an office desk with top mounted cameras and a projected tabletop display instead. A low-resolution top-mounted camera tracks a users fingertip movement on the desk surface. The user points and taps on a document to take a snapshot of it with another top-mounted high-resolution camera. The system then processes the information on the photographed page for further manipulation with the computer. As Wellner points out in his thesis, the video tracking algorithm used has some drawbacks since it only tracks motion. This algorithm is improved in the follow-up project *EnhancedDesk* [35], which features a similar setup, but enhanced input tracking and interaction metaphors. EnhancedDesk also investigates augmenting physical items like books with additional projected information. Items on the tabletop are tracked with a marker system developed by Rekimoto, which is also utilized for the *Augmented Surfaces* setup described in section 3.1.6.

3.1.3 I/O Bulb and Sensetable

Urp by Underkoffler and Ishii is a tangible workbench based on the *I/O Bulb* infrastructure [68]. *I/O Bulb* incorporates top-mounted projectors and an optical tracking system to augment surfaces and allow interaction with these surfaces through tangible objects. The *Urp* workbench uses this technology to provide urban planners with means to predict wind flow, shadows and reflections in a city through architectural models. *Sensetable* by Patten et al. derives from the *I/O Bulb* system [50]. Instead of optical tracking, *Sensetable* uses electromagnetic tracking to determine the position of up to 10 physical objects (called ‘pucks’) on the tabletop. A top-mounted projector displays applications onto the tabletop, where the pucks are used as input devices. Example applications include a business supply chain visualization and *Audiopad*, a tool to control a musical performance [51].

3.1.4 Roomware

Roomware is a research project at the *Fraunhofer GMD-IPSI* which investigates possibilities to embed computer interfaces into the architectural environment of a *cooperative building* [53]. This way, computers should enhance collaboration, but mentally disappear for the user. *Roomware* components include *DynaWall*, *CommChair*, *ConnecTable* and *InteracTable*. *InteracTable* is a meeting table with an integrated plasma display and a touch frame. The *Beach* software framework used on the *InteracTable* allows multiple document views and workspace rotation.

3.1.5 BlueSpace

BlueSpace is an *IBM* research project to create a next-generation workspace for increasing knowledge worker’s productivity [38]. While this workspace uses an LCD screen as a primary display, it incorporates projected peripheral displays to display urgent messages on the office desk. This way, users of the system can focus on their primary tasks without interruptions, but still receive important notifications. The *BlueSpace* project incorporates the *Everywhere Display* [52], a steerable projector combined with camera that transforms various surfaces like tabletops and walls into interactive displays on demand.

3.1.6 Augmented Surfaces

Augmented Surfaces [54] by Rekimoto and Saitoh allows users to exchange data between various different devices like notebook computers intuitively. The table of a meeting room represents a server. Users can copy files from their laptops to the server by dragging them onto the tabletop with their laptop cursors. The file content is then displayed on the tabletop and can be

moved with the cursors or a laser pointer. A similar data exchange metaphor is used the *Coeno* setup [23], where images, videos and 3D models can be modified on tablet PCs and then dragged onto the table for discussion.

3.1.7 DiamondTouch

DiamondTouch by Dietz and Leigh [11] is a system to determine touch input on a table surface by using a conductive tracking technology. Inside the table surface, a grid of wires transmits unique electric signals to different regions. When a user touches the surface, the signal flows through the user's body into a receiver. This way, the system is able to determine multiple user touch input simultaneously. Since the tracking surface is not transparent, *DiamondTouch* setups use ceiling-mounted projectors to display information on the tabletop.

3.1.8 Shared Design Space

Shared Design Space [24] by Haller et al. is a cooperative environment designed for brainstorming and discussion meetings. Users can create collaborative sketches on a digital tabletop with pen input and send them to an interactive wall where they are arranged along a time line. This setup is the first to utilize Anoto tracking technology for tabletop pen tracking. A follow up project, the *Conoto Graphics Tablet* by the Author enables single user pen input for Windows applications. Parts of the *Conoto* code are used for the *Virtual Graphics Tablet* application on the *PenTable* setup, which is described in section 7.5.

3.1.9 Microsoft Surface

Surface was introduced in May 2007 by Microsoft¹. It consists of a 30-inch rear projection display in a table-like form factor. The tabletop surface is multi-touch enabled and allows small groups of users to interact with a variety of applications for home, retail and entertainment settings [44].

3.1.10 Interactive environment-aware display bubbles

Interactive environment-aware display bubbles take into consideration that the rectangular shape of WIMP applications might not fit well on a tabletop display [8]. Cotting et al. propose an algorithm to warp the application windows onto user defined shapes called *display bubbles*. A top-mounted camera tracks the dot generated by a laser pointer for user interaction. Users define the shape of display bubbles by drawing their outline with the laser pointer. The display bubbles can be modified at any time using the

¹www.surface.com

laser pointer and a pie menu. The top-mounted camera also tracks physical objects on the table to avoid collisions with display bubbles.

3.1.11 Visual Interaction Platform (VIP)

VIP is a single user workspace for the early stages of architectural design [3]. It allows to fluid switching between direct tangible interaction on a tabletop display and WIMP interaction on a vertical display. Tasks that profit from two-handed interaction and direct pen input are carried out on the tabletop display. The vertical LCD display shows a standard WIMP GUI and acts as a communication area. *VIP* could be used as a multi-purpose workplace similar to the PenTable setup, but the tabletop interaction is primarily designed and therefore currently limited to architectural scenarios.

3.2 Peripheral and Ambient Displays

Wisneski et al. propose *ambient displays* to present digital content in a calm and undisturbing way by leveraging the users peripheral awareness [72]. The *ambientROOM* setup provides information about network load, past events and human activity awareness through sound displays and augmented walls and ceilings.

With *focus+context screens*, Baudisch et al. combine a small high-resolution display with a large low-resolution context display and investigate user performance in a comparative study [6]. Study participants were able to focus simultaneously on a primary work task and an additional monitoring task by using their peripheral vision.

The *Kimura* system combines focal and peripheral interactive displays for an enhanced office workplace [41]. The focal display is a monitor in front of the user, which displays the current task. The peripheral display is a large interactive wall projection, which presents information about background activities and context-aware notifications. This way, the system supports the peripheral awareness of such information and enhances user multitasking between different working contexts.

Morris investigates the benefits and challenges of peripheral tabletop displays and states that tables are appealing as peripheral displays because they are already part of the users environment and can easily be transformed into a display with a projector [47]. She describes an example of an ambient activity visualization displayed on the *AmbienTable* and formulates design suggestions to address challenges like orientation, shape, size and occlusion on peripheral tabletop displays.

3.3 Direct Pen Input

Elliott and Hearst conduct a comparative user-study with architectural design tasks carried out on a 19-inch display with mouse input, a 14-inch display with pen input and a 42-inch tabletop display with pen input [13]. Test participants complete an image sorting and a sketching task on all three setups. The results show that participants equally preferred the pen based setups to mouse input for the sketching task. For the image sorting task, the desk setup was the least preferred system and resulted in the highest sorting times and most mistakes. Surprisingly, the lower pixel density of the tabletop display was not perceived a problem. However, participants complained about pen parallax, which suggests that a different pen tracking technology and thinner display surface could improve performance on the tabletop display. According to Elliot and Hearst, the possibility to present peripheral information on the tabletop display should be further investigated and it could be useful to explore how combined displays of different size would work together as a system.

Accot and Zhai investigate user performance of pointing at a target with a pen based input device versus crossing it and propose an interface based on crossing instead of point-and-click [1]. Their findings are applied to *CrossY* by Apitz and Guimbretière, an alternative to the standard WIMP interface entirely based on crossing [4]. Examples of context menus developed especially for pen input include *FlowMenu* [21] and *Tracking Menus* [18]. Tracking Menus were designed to prevent round trips to tool palettes with the pen, but could also be used to move the menu when physical objects occlude menu options on the tabletop. While this type of context menu could be used for the PenTable GUI, it requires an input device, which supports a hover state. Since the tracking technology of PenTable does not support such a state, the application of Tracking menus to the setup was not further investigated.

The occlusion shadow of a user's hands on a direct input display can obstruct pop-up menu options. Hancock and Booth propose several strategies to improve the placement of such context menus [26]. These strategies determine the user's handedness to predict a possible occlusion and adapt the menu position on a tabletop display accordingly. However, the proposed strategies do not consider occlusion shadows created by physical objects. Therefore, different context menu metaphors have been developed for the PenTable setup, which are described in section 5.2.

Li et al. investigate various mode switching techniques in pen-based user interfaces [40]. Their findings suggest that pressing a separate button with the non-preferred hand offers the best performance for differentiating between input and gesture modes for pen interfaces. While the PenTable setup does not utilize pen gestures, this technique is applied to activate the context menu on the tabletop.

TractorBeam by Parker et al. uses a magnetically tracked pen to evaluate remote pointing on tabletop displays [49]. Custom metaphors are proposed to reach items from locations further away on the tabletop and to interact with items close to the user. The results of a pilot user study demonstrate that pointing is faster than stylus input to reach large distant targets and slower for small targets. Since reach is not a primary concern for the design of the PenTable setup and the chosen pen input device does not support pointing at a target in 3D, these findings have not been applied to the system.

Bumptop by Agarawala and Balakrishnan applies physics to the icons on the Windows desktop of a tablet PC [2]. The possibility to create piles and stacks of files enables a natural interaction with digital content. While physics were not applied to content on the PenTable setup, the proposed information browsing metaphors inspired the design of pMenus, which are described in 5.2.

3.4 Analysis

Current research projects utilize the horizontal layout of digital tabletops because of the following reasons:

- A digital tabletop display provides more room for simultaneous direct input than a vertical display.
- Group interaction benefits from a tabletop display, since people sitting around it face each other.
- Metaphors for sharing or exchanging data between users may seem more consistent on a tabletop display.
- Physical objects utilized as computer interfaces are easier to position and move on a tabletop.
- Working with direct input devices like stylus or touch input is less tiring on a tabletop than on a vertical display.
- A digital tabletop can be integrated into tables already present in the users environment for an unobtrusive interface.

Digital tabletop setups have been developed for over 15 years and numerous papers have been published about the benefits of such interfaces. However, no *multi-purpose* workplace utilizing a digital tabletop is commercially available at the time of writing. We have identified hardware problems, space requirements and limited versatility as the main reasons.

Hardware: Many research projects use tracking systems, which are not stable enough, or too imprecise for daily use. Most tabletop displays do not provide the color consistency and pixel density which professional users expect from their primary work screen. The hardware for tabletop screens and input devices is often custom built and therefore expensive.

Space Requirements: Office workspace is limited and standard PC input devices are designed to use little space. If the digital tabletop GUI and input devices require an empty table surface to function, the workplace can not be used like a regular office desk. It therefore needs additional space.

Versatility: A reason the PC is so widespread is its large set of features and the versatility of its hardware interfaces. Most digital tabletop research projects are designed for a single specific task. The software and hardware interfaces of such a setup therefore often lack the versatility of standard PC interfaces.

Chapter 4

PenTable

Instead of replacing standard PC interfaces with a digital tabletop, PenTable combines them. LCD display, keyboard and mouse are used as primary interfaces for most tasks, while the digital tabletop acts as a peripheral display. For certain tasks, however, the tabletop is also utilized as a main input and output device. These tasks include sketching, taking notes and accessing program function shortcuts.

4.1 Hardware Interfaces

The PenTable setup combines two different displays, a vertical LCD display and a large projected tabletop display. Their attributes are compared in Table 4.1. Due to its high pixel density and color consistency, the vertical LCD display is used as the primary work display for most tasks (cf. Figure 4.1 (a)). While the user focuses on the LCD display, size and orientation of the tabletop display are utilized to presents ambient information. When a pen based interface is preferable for tasks like sketching, the user focus shifts to the tabletop display and it is used as a primary input device (cf. Figure 4.1 (b)). Each display has its own pointing device to avoid user confusion. The mouse controls the cursor on the LCD display while the digital pen is used as the pointing device for the tabletop display. This clear distinction allows a large-display user experience without the usability problems described by Robertson et al. [55].

4.2 Functionality

As described in the previous section, the user interacts with the LCD display like on a regular PC setup. The applications on the tabletop display can be compared to *widgets* in *OS X Dashboard*¹ or *gadgets* in the *Windows*

¹www.apple.com/macosx



Figure 4.1: Pentable interfaces. Standard PC interfaces like keyboard, mouse and LCD screen are used for most tasks, while the tabletop display presents ambient information (a). For tasks requiring pen-based interaction, the user focus shifts to the tabletop interfaces (b).

Table 4.1: Comparison of the PenTable tabletop display and LCD screen.

	Tabletop display	LCD screen
Size	Large (40" to 80")	Small (17" to 20")
Physical Orientation	Horizontal	Vertical
Pixel Density	Low (< 30 ppi)	High (> 72 ppi)
Color Consistency	Low	High
Input Device	Direct (Pen)	Indirect (Mouse)

*Vista Sidebar*². But unlike the Dashboard, the PenTable applications are always displayed and therefore can be used to present ambient information. The advantage compared to the Sidebar is that the PenTable applications do not interfere with the regular WIMP GUI on the task display, so it remains uncluttered. Another advantage to the Dashboard widgets and Sidebar gadgets is the direct pen input, which allows customize the look of applications and to sketch content with little effort.

Some of the PenTable applications provide ambient information like the current time in distant locations (cf. Figure 4.2 (a)) and event reminders (cf. Figure 4.2 (b)). Other applications make utilize the pen input to sketch on notes (cf. Figure 4.2 (c)) and create program buttons (cf. Figure 4.2 (d)). All of these items are created, used and modified directly on the tabletop with a digital pen.

²www.microsoft.com/windows/products/windowsvista

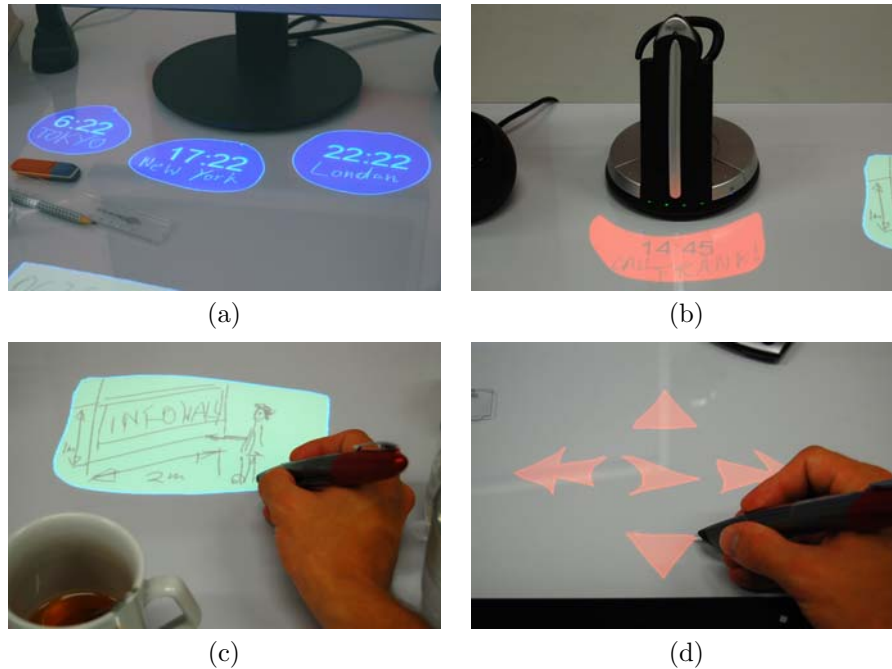


Figure 4.2: PenTable functionality. The world time clocks display the time in distant locations (a). Next to the headset, the user has placed a reminder for a call (b). A sketch on a virtual note (c). User-drawn media player controls (d).

4.3 Features

The introduction of this thesis describes some drawbacks of standard PC interfaces and section 3.4 some shortcomings of current tabletop research projects. They can be summarized as current PC workplaces lacking natural interfaces for certain tasks, and tabletop setups lacking the feature set and versatility of standard PCs. PenTable is designed to combine the advantages of both worlds by offering the features of a standard PC and an intuitive tabletop interface. The main features of PenTable are robust and inexpensive hardware, no additional space requirements and high versatility.

- **Robust and inexpensive hardware:** The setup uses only commercially available hardware. Therefore, the interfaces are tested for daily use and costs are kept at a minimum. Since tabletop projection is used, no electrical components are built into the table, so it is unaffected by spilled liquids or mechanical stress. Because the stylus tracking is built into the digital pen, it is not disturbed by other PC interfaces, physical objects or lighting conditions.

- **Space Requirements:** Not only the hardware but also the graphical user interface can be used while the tabletop is cluttered with physical objects. The space requirements of the PenTable setup are similar to those of a standard PC workplace because the tabletop can still be used as a place to hold physical objects.
- **Expressiveness:** The PenTable system does not sacrifice the versatility of a desktop PC because of the tabletop interface. Any Windows software can be installed and used like on a standard PC workplace. The tabletop applications are designed to extend the functionality or ease of use of such software.

In addition to these features, the PenTable setup is designed to be non-intrusive, which means the PC can be used without the additional tabletop interface. In that case, users would probably not even notice a difference to a standard PC workplace.

Chapter 5

Tabletop GUI

The WIMP GUI has been developed for mouse input on a vertical LCD screen like the one shown in Figure 5.1 (a). The tabletop display used for PenTable differs from such a screen in size, shape, orientation and input device, as Figure 5.1 (b) depicts. These characteristics of the tabletop display require a special graphical user interface. WIMP application windows and icons are replaced by sketched widgets (*sWidgets*) and instead of standard pop-up menus, user-drawn path menus (*pMenus*) are used.

5.1 *sWidgets* Instead of Windows and Icons

When a WIMP application window is opened on the LCD display, it is rendered with a predefined size and position. If these do not fit user requirements, the window can be resized and repositioned with the mouse cursor. A window is moved by dragging the title bar and resized by dragging the borders to another position. However, on a tabletop display with direct pen input the window title bar and borders can be hard to reach. There-



Figure 5.1: The PenTable LCD display shows a standard WIMP GUI (a), while the characteristics of the tabletop display require a special GUI (b).



Figure 5.2: Standard WIMP application window used on the PenTable tabletop display.

fore, a possibility to define the window size and position prior to opening it can increase the usability of the tabletop GUI significantly. An additional problem on the tabletop display is physical objects on the tabletop. Due to these objects, the available screen real estate can vary in size and shape. Figure 5.2 depicts a typical WIMP application window being used on the PenTable tabletop to demonstrate the problem. If the window would not be constrained to a rectangular shape, it could adapt better to the available screen real estate.

For this reason, sketched widgets (*sWidgets*) replace the window metaphor on the PenTable tabletop display. A *sWidget* shape is defined by sketching its outline and can therefore be easily adapted to the available tabletop screen real estate and user preferences. This possibility of defining a *sWidget* shape can not only be used to adapt it to the available display space, but also to create a symbolic representation of the *sWidget* functionality. Figure 4.2(d) depicts a media player, where the shape of *sWidget* controls reflect their function.

5.1.1 Creating *sWidgets*

Instead of clicking on an icon to open a rectangular window like in a standard GUI, a user sketches the outline of the *sWidget* onto the table. Figure 5.3 depicts a scenario, where the user is creating a *sWidget*. If a drawn stroke is recognized as a closed shape, the user receives visual feedback and the shape remains on the tabletop when the pen is lifted. In the next step, the

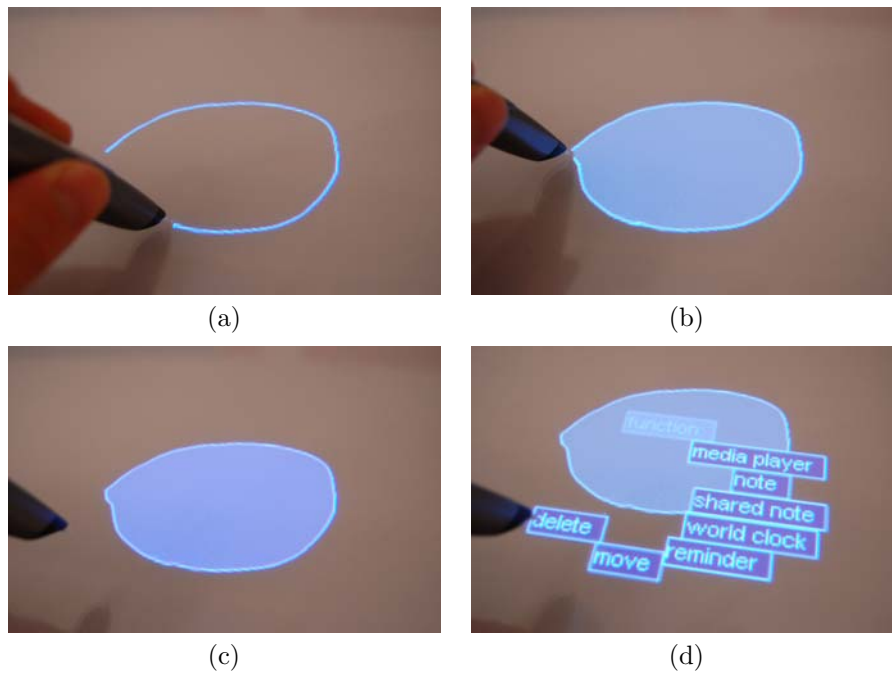


Figure 5.3: To create a sWidget, its shape outline is sketched with the pen (a). When a closed shape is recognized, the system highlights it (b). It remains on the display after the pen is lifted (c). The sWidget type is defined with the context menu (d).

user can open a control menu. Finally, by selecting the corresponding menu item, the sWidget type is chosen.

5.1.2 Modifying sWidgets

A sWidget can be changed, saved, moved or erased at any time with its context menu. To differentiate between using a widget and accessing its context menu, the user presses the control key on the keyboard while tapping on the widget with the pen. Pressing a key with the non preferred hand has been recommended as a pen mode switching technique by Li et al. after conducting a comparative study with different alternatives [40].

5.2 pMenus Instead of Menus

All complex program functionality on the tabletop display is accessed with context menus. This way, menu options are always displayed within reach of the user. When a standard pop-up context menu is opened, it displays all menu options at once. The size, layout and position of such a pop-up menu can therefore cause it to overlap with physical objects nearby. Figure 5.4



Figure 5.4: Drop-down menus (a) and pie menus (b) become hard to see or even unusable when overlapping with physical objects on a tabletop display.

depicts a scenario where both a pie menu and a rectangular menu are partly occluded due to overlapping. To be able to view all menu items, the menu or the physical object has to be moved. Either one of these actions disrupts the workflow and possibly leads to a suboptimal and undesired workspace arrangement.

Previous research to avoid menu occlusions produces very good results when dealing only with the user’s hand [26]. But possible occlusion shadows created by physical objects on a tabletop are harder to determine by the system. Also, these occlusions can be more disrupting to the user, since objects like full coffee cups are more cumbersome to move than a hand.

To avoid context menu occlusions, the setup could try to determine the size and position of all physical objects on the tabletop with a tracking system (e.g. external camera tracking). It could then adapt the menu position and appearance accordingly. A disadvantage of this approach is the added technical overhead. Another problem is the lack of comprehensible consistency in the menu placement decision. If a menu is lacking space at the desired location, the system has to decide to either change its size and shape or to render it at another position altogether. Since these decisions are made by the system, the user never really knows where the menu will be displayed next.

The PenTable GUI uses a different approach, since it does not predict possible occlusions, but lets the user decide the menu size and shape. Instead of the context menu popping up after user input, it appears along a user-drawn path (cf. Figure 5.5). This menu metaphor is called *pMenu* and derives from the *layers-as-a-stack-of-cards* analogy [42], which is based on the idea of representing information layers as individual cards of a card deck. These cards can be positioned as individual items. The menu option of *pMenus* are represented by such individual items and placed along a user-drawn path.



Figure 5.5: The menu items of a pMenu are placed along a user-drawn path to avoid occluding objects on the tabletop.

The advantages of placing the menu options along a user-drawn path instead of arranging them in a predefined or system-decided shape are:

1. Menu occlusions can be easily avoided by drawing a path around them.
2. Since menu placement and shape are user-defined, they are more comprehensible than decisions made by the system.

5.2.1 User Interaction

Only the root element of a pMenu appears when the pen tip touches the display surface. The rest of the menu unfolds as the pen moves on the display. The individual menu items are placed along a path defined by the pen tip, and stay on their position when the pen is lifted. The user can then select an item by tapping on it or close the menu by tapping on the root item.

5.2.2 pMenu Metaphors

Many different possibilities exist to place the individual menu items while the path is drawn. The following sections describe four possibilities: the *Fan Out Menu*, the *Card Deck Menu*, the *Pearl String Menu* and the *Trail Menu*. Although only the *Card Deck Menu* metaphor is used for the PenTable

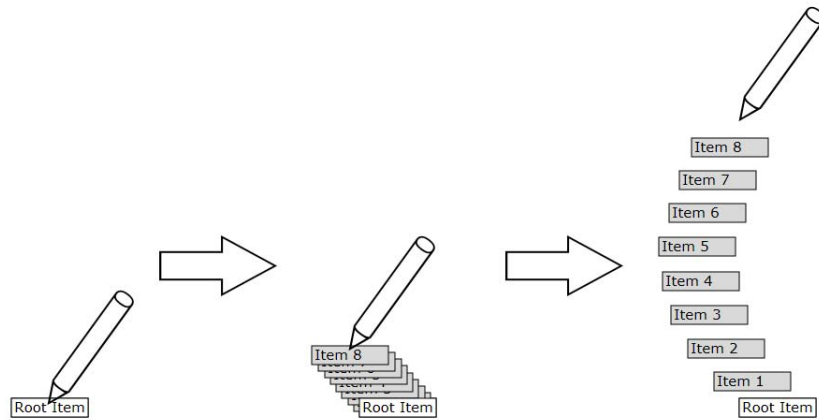


Figure 5.6: Fan Out Menu. While the pen tip moves, all menu items are spread simultaneously on the pen path.

system, one of the other three metaphor could be more appropriate for a different setup.

Fan Out Menu

This metaphor is similar to the approach described by Agarawala and Balakrishnan [2] to display information by fanning it out along the pen path. All of the menu items are positioned on the path between the start and current end point, with an even path distance between each items center point. While the path is drawn, the path distance between the items grows, so they appear to fan out (cf. Figure 5.6).

An advantage of this metaphor is that since all menu items are spread out along the path simultaneously, the total number of items is immediately visible to the user. Therefore the path length needed to display all menu items can be predicted. A disadvantage to the other user-drawn menus is that the items partly occlude each other until they are completely spread out. Therefore, it is hard to select an individual item before all are spread out.

Card Deck Menu

The Card Deck Menu behaves similar to a deck of playing cards with individual cards taken from the deck and placed behind each other on the table. When the pen tip touches the surface, the root element is displayed. While the pen moves on the surface, the distance between each new pen point and the previous menu item is evaluated. A new menu item is placed with its

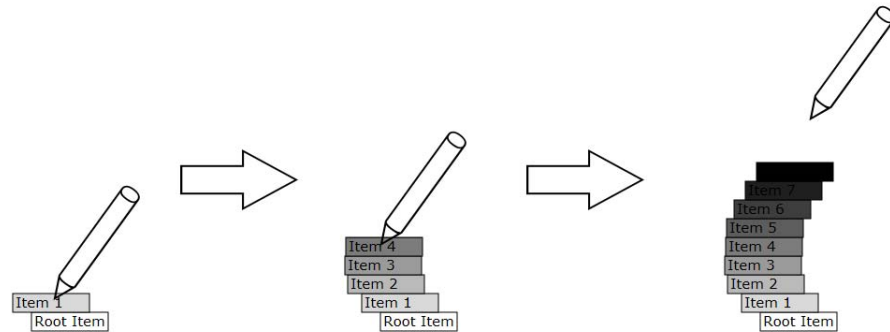


Figure 5.7: Card Deck Menu. While the pen path is defined, menu items are placed along this path after each other. Menu items shown darker in the figure appear later.

center at the pen tip if it does not occlude the previous menu item. In contrast to the fan out menu, each menu item stays at the position it is placed. After all items are placed, moving the pen has no further visible effect on the menu (cf. Figure 5.7).

An advantage of the card deck menu is the possibility to place a new menu item on top of any item other than the immediate predecessor to save space. This metaphor also allows selecting an item immediately after it is displayed without having to place the remaining items.

Pearl String Menu

Like in the Card Deck Menu the menu root element is displayed when the pen tip touches the surface. However, while the pen moves, the menu items follow the position of the pen tip. Subsequent menu items are placed behind each other along the path from the pen tip to the root item. Each item is placed to not occlude its predecessor. As soon as all menu items are visible, moving the pen has no further effect on the menu (cf. Figure 5.8).

An advantage of the pearl string menu is that while the menu is opened, the menu item order relative to the pen tip does not change. When adapting that order to user behavior similar to *split menus* [16, 58], important items always stay close to the pen tip while the menu is drawn. This behavior could lead to a better menu performance.

Trail Menu

The behavior of the Trail Menu is similar to the Pearl String Menu. But while the appearance of the Pearl String Menu does not change anymore

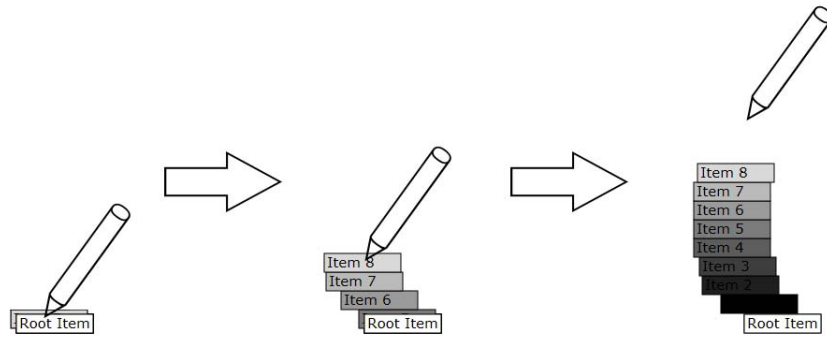


Figure 5.8: Pearl String Menu. While the pen path is defined, menu items seem to be pulled out from the root element along the path. Menu items shown darker in the figure appear later.

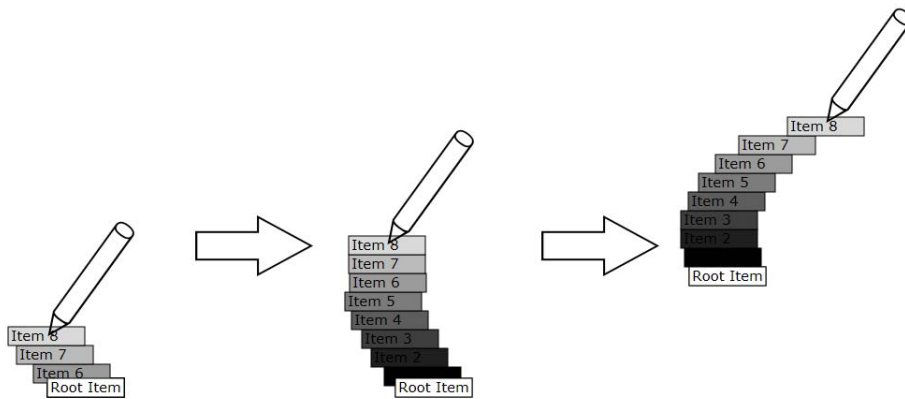


Figure 5.9: Trail Menu. While the pen tip moves, the menu follows it like a tail. Menu items shown darker in the figure appear later.

after all items are visible, the items of the Trail Menu follow the pen tip like a tail as long as it is moved (cf. Figure 5.9).

An advantage of this metaphor is the possibility to change the menu position and layout even after all menu items are displayed. While the initial menu placement is critical to the positioning of all other menus, the trail menu can be moved until a desired menu position is found.

Chapter 6

Implementation

6.1 Hardware Setup

Figure 6.1 depicts the hardware setup, which consists of a standard PC with a keyboard, monitor and mouse. In addition, a projector is mounted on top of the tabletop, converting it into a large scale display. The PC used in a first version of the setup consisted of a *Intel Pentium 4* processor with 2.5 GHz, 1 GB of RAM and a *GeForce 6800* graphics card with dual display outputs. Because more VGA outputs were required for the added projectors of another test setup, it used a PC with two *GeForce 7800* graphics cards, which had an *Athlon 64 X2 3800+* processor and 2GB of RAM.

6.1.1 Tabletop Display

A ceiling mounted projector was chosen for the tabletop display. The main reasons were an easy assembly and an unaltered table form factor, which would not have been possible with a system based on a rear-projection display or an LCD panel. In addition, the Anoto tracking technology only works on projection-based surfaces at the moment.

Different projectors were used for the various PenTable prototypes. The *Sanyo PLC-XU86* proved to be a good choice because of its small throw ratio of only 1,26. Disadvantages of this projector are the lack of a lens-shift and a native resolution of 1024×768 pixels, which leads to a small pixel density of the tabletop display. A second version of the PenTable system utilized an arrangement of three *Optoma EP 1690* projectors with a native resolution of 1280×768 pixels to create a 67×35 -inch tabletop display with resolution of 2304×1024 pixels.

6.1.2 Pen Input

Anoto tracking is used for direct pen input on the table surface. The features and background information of the Anoto tracking technology are described

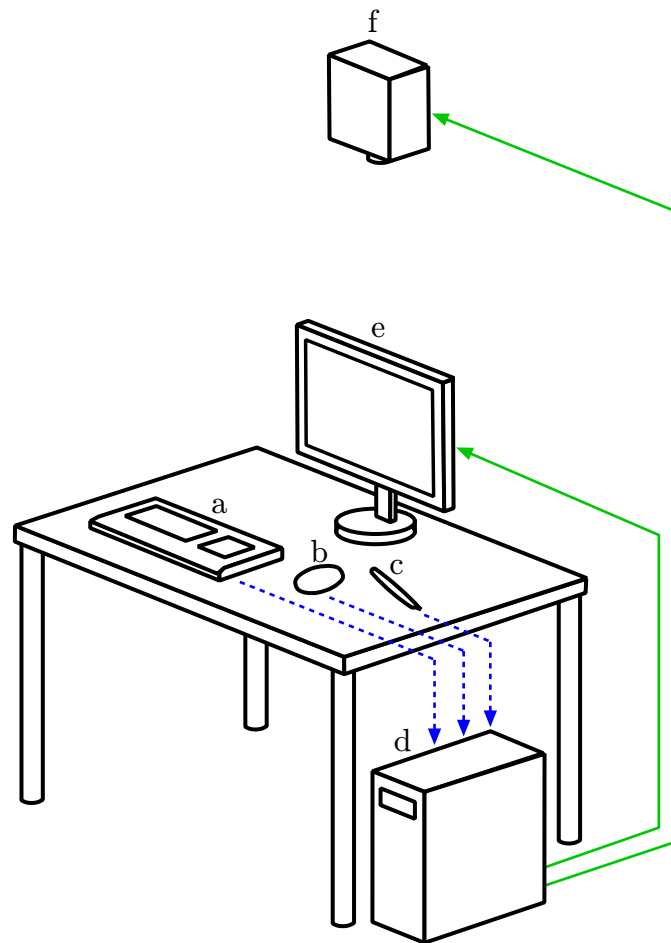


Figure 6.1: The input devices keyboard (a), mouse (b), and digital pen (c) are wirelessly connected to a PC (d) under the table. The PC's graphics card outputs are connected to an LCD display (e) and a ceiling-mounted projector (f).

in section 2.1.1. The Anoto pattern was printed on an A0 sized sheet of paper with a 600dpi ink jet plotter and protected with a scratch resistant acrylic surface. Some test setups used an area larger than A0, therefore multiple paper sheets with unique pattern were printed and aligned next to each other on the table surface. The pen tracking does not work at the borders of these sheets, which disturbs user interaction. The Anoto pen used for all setups is a *Maxell digital pen dp-201 R4.1*. An *Ednet Class 1 Version 2.0 EDR* dongle is used for the wireless connection between pen and PC.

6.2 Software Implementation

The PenTable prototype was programmed in C# with GDI+ for rendering. The current setup is running on Windows XP with Service Pack 2 and .Net Framework 3.0 installed.

Microsoft COM (Component Object Model) technology¹ is utilized to communicate between the PenTable applications and Windows programs like Apple iTunes and Microsoft OneNote.

To use the digital pen as a direct input device, it is necessary to map each tracked point on the Anoto pattern directly to a corresponding point on the projected tabletop display. The homography matrix needed for this mapping is determined with an algorithm taken from [48, p. 127].

¹<http://www.microsoft.com/com>

Chapter 7

Example Applications

A number of example applications have been developed to demonstrate possible usage scenarios of the digital tabletop. While some of the applications use the tabletop as a direct input device, others utilize it as a peripheral to display ambient information. All tabletop applications use sWidgets to define their shape, size and position. To change an applications properties, a pMenu is opened by pressing the left control key on the keyboard and drawing the menu path from the sWidget.

7.1 Note Taking

The note taking application allows users to sketch virtual notes on the table surface. These notes can be used to display various information on the tabletop displays or to create a sketch and insert it into a document on the primary LCD screen. The note content can be sent to Microsoft OneNote 2007¹ and then copied to other Windows applications either as a bitmap or as vector data. Figure 7.1 depicts the process of taking a note and sending it to OneNote:

1. After a new sWidget is drawn, “note” is selected from its context menu.
2. The sWidget now turns yellow and can be annotated with content.
3. When “send to OneNote” is selected from the notes context menu, the note disappears from the tabletop.
4. The content is then inserted into a new page in Microsoft OneNote.

¹<http://www.onenote.com>

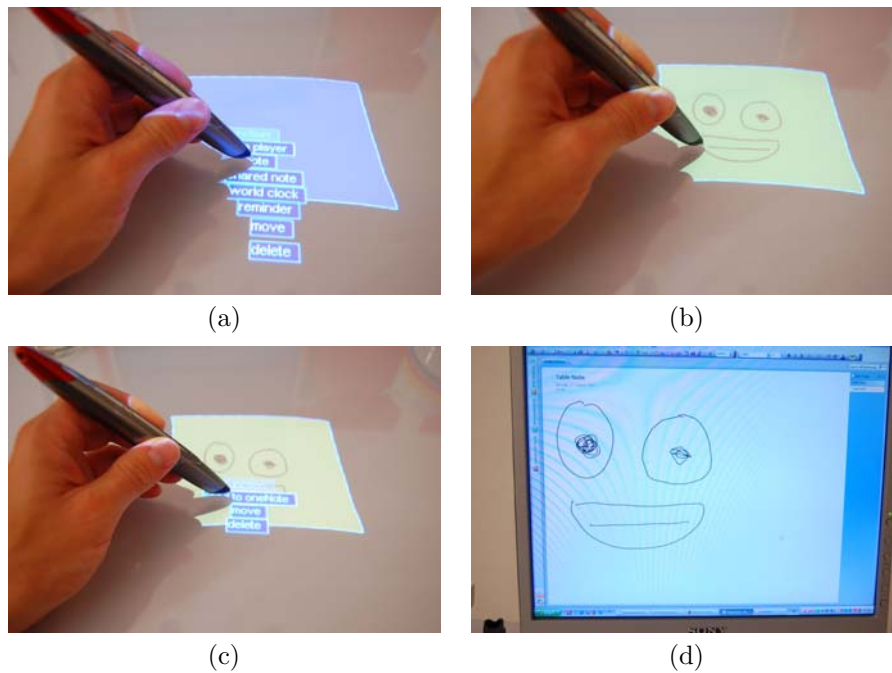


Figure 7.1: Creating and using a note. After the sWidget shape is defined, “note” is selected from its context menu (a). The shape background turns yellow and can be annotated with the digital pen (b). The note content is sent to OneNote with its context menu (c). The note disappears from the tabletop and its content appears on a new page OneNote (d).

7.2 World Clock

The world clock displays the current time with a user defined offset in a sWidget. This functionality can be used to display the time of a different distant location in the periphery of the user. The world clock can be annotated with additional content to customize its look or to inform the user of the time zone location, as Figure 7.2 depicts. The world clock is created and modified in the following way:

1. After a new sWidget is drawn, “create clock” is selected from its context menu.
2. The sWidget turns blue and the current Windows time is displayed inside.
3. The sWidget can be annotated with additional content like the name of the time zone.
4. Using the sWidget context menu, a time offset can be set.

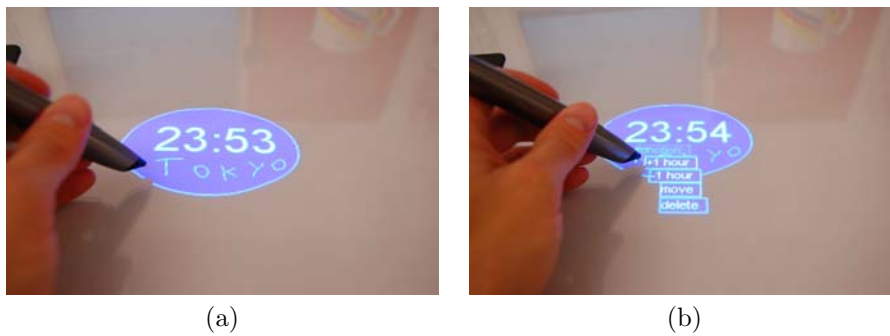


Figure 7.2: Modifying the world clock. After the world clock is created, the area around the current time can be annotated with the pen (a). The desired time zone is set with the clocks context menu (b).

7.3 Media Player Controls

This application allows users to draw buttons on the table surface, which can be linked to the playback functionality of Apple iTunes². The controls on the tabletop act as shortcuts, so the user does not have to open the iTunes interface on the LCD screen to skip tracks or change the volume. The possibility to define the sWidget shape can be used extensively to create an iconic representation of each playback functionality and to adjust the appearance of the player to individual user preferences. Figure 7.3 depicts the creation and usage of a media player button to control iTunes playback:

1. After a new sWidget is sketched, “media player” is selected from the context menu.
2. Then, the playback functionality is linked to the control using the context menu.
3. When the control is tapped with the pen, the associated playback function is executed in iTunes.

7.4 Reminder

The reminder is a tool, which can be set to notify the user of an event at a certain time. After its creation, the reminder displays the time of the next event, which be annotated with additional information. During the last ten minutes before the event, the background color changes from green to red, as Figure 7.4 depicts. This color information allows users to keep track of the remaining time through their peripheral vision without having to shift their focus to the tabletop display.

²<http://www.itunes.com>

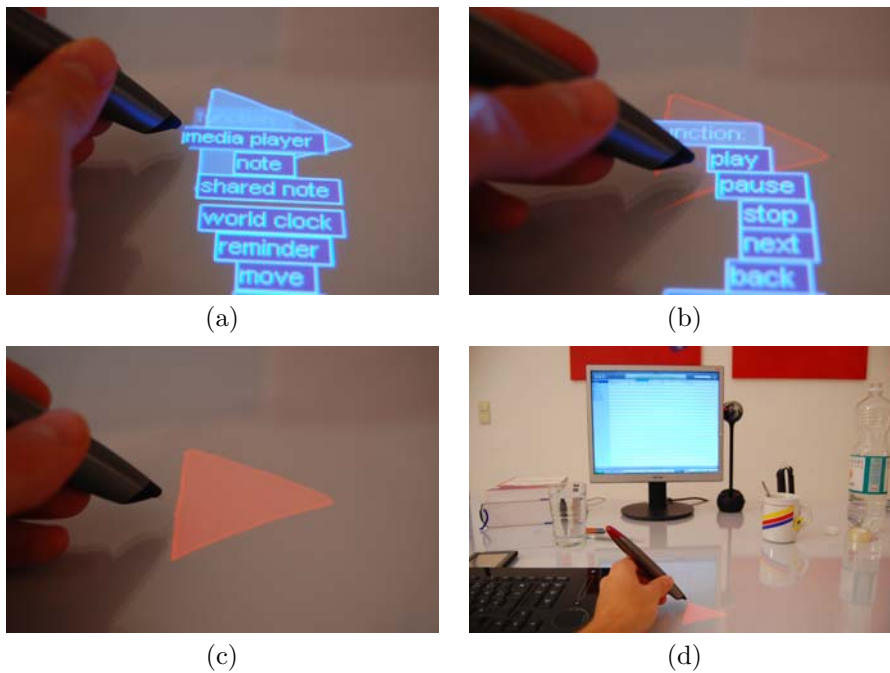


Figure 7.3: Creating and using a media player control. After the sWidget shape is defined, “media player” is selected from its context menu (a). The shape background turns dark red to indicate an unfinished media player control. Now the context menu can be opened again to select the specific playback functionality (b). The background turns red and the media player control can be used (c). When the control is tapped with the pen, the associated playback functionality is executed in iTunes (d).

7.5 Virtual Graphics Tablet

The virtual graphics tablet functionality is similar to a commercially available electromagnetic graphics tablet. It is used for indirect pen input on the LCD screen, when a user wants to use a pen device for programs like Adobe Photoshop. The advantage to a physical graphics tablet is the adjustable size of the virtual graphics tablet, so it only uses as much space as needed on the tabletop. The following process describes the creation and interaction with the virtual graphics tablet:

1. After a new sWidget is drawn, ‘create graphics tablet’ is selected from its context menu.
2. The sWidget turns green and the graphics tablet widget can be used to control the mouse cursor on the screen.

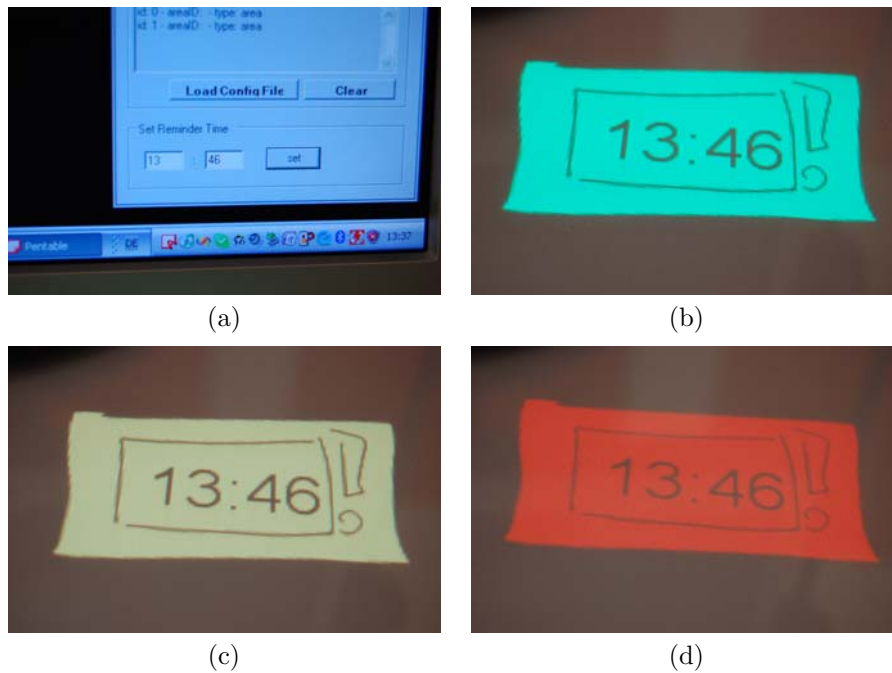


Figure 7.4: Using the reminder. The event time is defined in a form displayed on the primary LCD screen (a). Up to ten minutes before the event, the reminder background color is green (b), it then turns to yellow 5 minutes before the event (c), and finally to red when the event approaches (d).

Technical Implementation: The Maxell digital pen does not support a hover state and it has no buttons to differentiate between left and right click, a crucial feature for mouse input in many Windows applications. Therefore, the pressure information from the pen tip is utilized to differentiate between these commands.

When the pen touches the tracking surface with little pressure applied, the mouse cursor on the screen is moved³, but no left mouse click is simulated. As soon as the pen pressure exceeds a certain threshold, the *left mouse button down*-event is sent. The *left mouse button up*-event is sent after the pen is lifted from the tracking surface.

To simulate a right mouse click, a time limit and a move threshold is used. When a *left mouse button down*-event would be called, but the pen movement does not exceed a certain threshold until a predefined time limit is reached, a right mouse click is simulated as soon as the pen is lifted. If the pen is lifted or moved before the time limit is reached, a normal *left mouse button down*-event is sent.

³All mouse events are generated with the `SendInput()` function in `User32.dll`.

Chapter 8

Evaluation

Two user-studies were conducted with parts of the PenTable tabletop GUI. The first study was an informal user study to receive feedback about different possibilities to create sWidgets. The second study compared four different pMenus with standard pie and pop-up menus on a cluttered tabletop. The overall PenTable setup was tried out by different participants at various occasions. Section 8.3 describes the feedback about the hardware design and PenTable applications.

8.1 sWidget Informal User Study

The current mode of creating sWidgets was developed after receiving user feedback from an informal user study with 6 participants at the Aalborg University Copenhagen. Figure 8.1 depicts the test setup. The participants were presented with 3 different way to create sWidgets on the tabletop:

1. Tapping a virtual button on the tabletop with the pen to create a rectangular sWidget with a predefined size (similar to opening a window in Microsoft Windows XP).
2. Tapping a physical item with the pen to open a rectangular sWidget.
3. Drawing an outline to define size and shape of the sWidget.

For each way, participants were presented with the possibility to select the sWidget type from a context menu or to write a character to select the type into the shape. Users expressed a strong preference for creating sWidgets by drawing their outline and then selecting their type with a context menu, therefore this way was implemented.



Figure 8.1: The test setup at the Aalborg University Copenhagen, which was used for the sWidget user-study.

8.2 pMenu User Study

8.2.1 Introduction

To compare the performance of pMenus with conventional rectangular pop-up and pie menus, a user study was conducted. The design and results of this study are also going to be published in the paper “Improving Menu Interaction for Cluttered Tabletop Setups with User-Drawn Path Menus”.

8.2.2 Hypothesis

According to the Hick-Hyman Law [27] [31] and to Fitts’ Law [17], the pMenus should be less efficient than traditional menus. One reason is the time required by users to draw the path for the user-menu, which takes longer than a simple click, needed for the pie/pop-up menu. The second reason is the varying distance from target to the root-item, according to the free space available on the display. Both Hick-Hyman and Fitts’ Laws are developed for menus that are visible immediately. Thus, no cluttering object is occluding the menu. Summarizing, we formulated the following hypotheses:

- Hypothesis 1: We expected better performance (in time and errors) for the pie menu and the pop-up menu compared to our proposed menus, while working on an empty table.
- Hypothesis 2: We expected better performance (in time and errors) for our proposed menus while working on a cluttered table. We also expected that participants will subjectively prefer our menus under this condition.

Based on these hypotheses, we measured the performance, the error rate provided by the task outcome, and the subjective measures (user opinions).

8.2.3 Experiment

Apparatus

We conducted a laboratory user study where users stand in front of a horizontal, rear-projected table (112×85 cm). The resolution of the projected display was 1024×768 pixels (cf. Figure 8.2(a)).

The tracking of the stylus is realized by using a large Anoto pattern printed on a rear-projection foil in conjunction with a *Maxell dp-201* digital pen. The key features of the Anoto tracking technology are listed in section 2.1.1. The Anoto pattern is clamped in-between two acryl panels to provide a stable and robust surface while protecting the pattern from scratches. The tabletop was accompanied by a 19-inch LCD screen. All menus for the

study were implemented in C# and rendered with Windows Presentation Foundation (WPF).

Participants

12 volunteers (8 males and 4 females), aged between 23 and 38 participated in the experiment. 11 participants were right handed and controlled the pen with their right hand. All participants were frequent computer users and had experience with Windows. Eight participants had previously worked with digital tabletop systems. 11 of them had already pen- and/or touch-based interface experience (e.g. tablet PC).

Task

For our experiment, users were presented with an item name on the LCD display next to the table. The users were instructed to open a menu on the tabletop and select the displayed item name as quickly and accurately as possible. The menu could be opened by either tapping the empty table surface for traditional menus, or drawing a path for user-drawn menus. Participants were asked to complete a series of five menu selections under three different conditions using six different menu layouts. Each menu item was 80×20 pixels, with the same size being used throughout the study. The Pie Menu had a radius of 100 pixels.

Our software logged all pen events and measured the time to complete the task from the initial display on the LCD display until the final selection. Whenever the participant selected a wrong menu item, an error was logged in our software, as was moving an occluding item.

Conditions

We used three different conditions for our experiment:

1. Empty table: users performed the task on an empty table (cf. Figure 8.2(a)).
2. Obscured table with movable content: under this condition the tabletop was cluttered with digital content to simulate physical objects. 36 randomly placed and rotated white rectangles were used to simulate paper, which occluded more than 50 % of the display. The participant had the possibility to move the content with the digital pen if it occluded an underlying menu, which always appeared on the bottom layer of the table (cf. Figure 8.2(b)).
3. Obscured table with non-movable content: in contrast to condition 2, the digital content could not be moved (cf. Figure 8.2(b)).



Figure 8.2: Empty tabletop (a) obscured tabletop (b).

Procedure and Design

A repeated measure within-participant design was used in our user study. The order of presentation of the six different menus was counterbalanced among participants. All users were presented with the same content. We also changed the order of the items appearing in the different menu categories to avoid a learning effect. The presentation (position) of additional content for both conditions two and three was randomized.

All participants had a short block of practice trials before each test session. Each condition lasted about 7 minutes. Participants took short breaks after every condition - an experimental session lasted about 30 minutes. Finally, users completed a post-experiment user preference questionnaire. Summarizing, the total number of trials can be computed as follows:

$12 \text{ participants} \times 5 \text{ trials} \times 6 \text{ menu types} \times 3 \text{ conditions} = 1,890 \text{ trials}$ in total.

The dependent variables measured were the time the users took for the overall trial (sometimes, users had to click the menu to different places to see all menu items), the selection time and the error rate and how many times occluding objects were moved until a menu selection was made.

8.2.4 Results

Performance Analysis

Figure 8.3 depicts the overall time participants used for the six menus under the three conditions. The overall time is the duration from displaying the task word on the LCD screen until the final selection of an item in a menu (this includes the time it takes to find the adequate menu position). An ANOVA of the collected data did not show a significant difference under the first two conditions. We also did not find any difference between the user-drawn menus and pie menus (although users had to draw the path menu

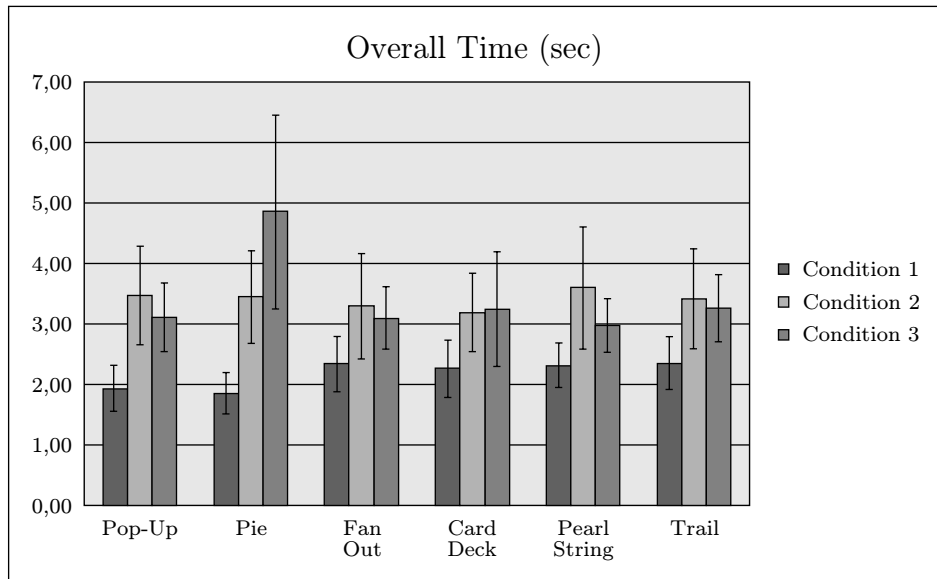


Figure 8.3: Overall time for each menu under the three different conditions.

which usually should take longer than just a single pen click).

We were surprised about the performance under condition 2, where the participants had the possibility to move the content. Although users had to move obscuring objects before being able to view the pop-up menus under optimal conditions, they were not slower than sketching the path for the user-drawn menu.

The overall time for the different menus under condition three was different: The Pearl String Menu was the fastest (mean 2.98, SD 0.45), followed by the Fan Out Menu (mean 3.11, SD 0.51), the Pop-Up Menu (mean 3.12, SD 0.57), the Card Deck Menu (mean 3.26, SD 0.95), Trail Menu (mean 3.27, SD 0.55), and the Pie Menu (mean 4.87, SD 1.59). We observed that users took longer using the Trail Menu, because most of them still moved the menu until they noticed that all menu items were already displayed. Participants had most problems with the Pie Menu under the third condition. The maximum time was 15.88 seconds using the Pie Menu for selecting one item. A repeated measures analysis of variance showed a high significant difference between the overall time under the third condition ($F_{11,60} = 12,316, p < 0,0001$).

Finding an adequate menu position was easier under condition 1 and 2 (cf. Figure 8.4). We found no significant difference between the times for finding the position for the six menus. In condition 3, however, it was sometimes impossible (especially for the Pie Menu) to find an adequate free position. We found a significant difference between the six menus under

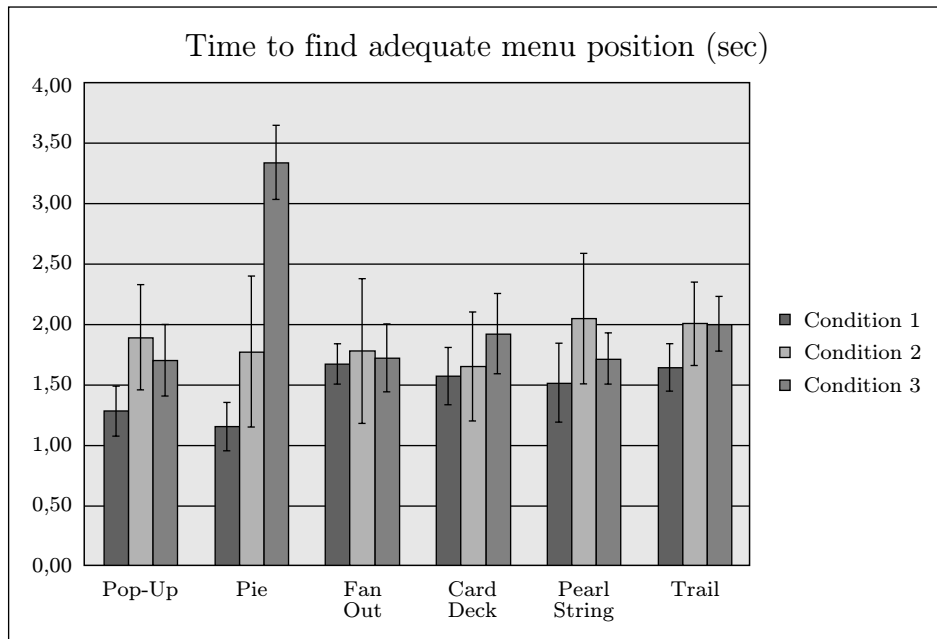


Figure 8.4: Time required to find an adequate position for the menu.

condition 3 ($F_{11,60} = 5,83, p < 0,001$).

The most extreme time difference was measured using the Pie Menu under the three different conditions. Due to the large radius of the Pie Menu, users could not find an adequate position on the table, which resulted in an average time of approximately 3.5 sec.

Figure 8.5 shows the average of moved objects under the second condition. Participants had to move more objects using the Pie Menu and the Pop-Up Menu. There was a significant difference between the Pop-Up/Pie Menu and the other menus ($F_{11,60} = 6,62, p < 0,001$), but no difference between the Fan Out Menu, the Card Deck Menu, the Pearl String Menu, and the Trail Menu. Only one person never moved any objects, even when given the possibility to do so under the second condition.

Figure 8.6 depicts how often participants had to open the menu to get the adequate position for selecting the target item.

Analysis of variance of counting the opened menus showed a high significant main effect for all menu-types ($F_{11,60} = 10,864, p < 0,0001$) under condition 3, where participants had no possibility to move the cluttering digital content. A post-hoc Tukey HSD test indicated that this was the only significant difference. No differences could be found for all menu types under condition 1 and 2. One person had to open the Pie Menu 12 times before getting able to select the correct item.

The time for selecting the menu item after having found an adequate

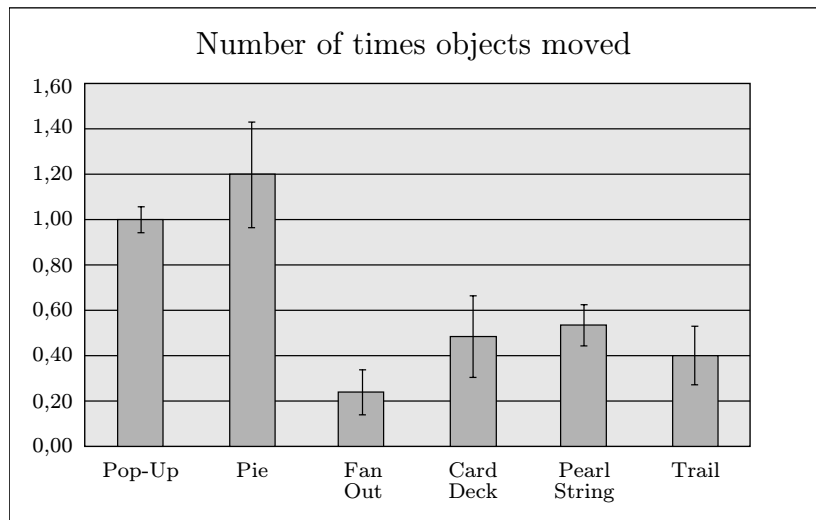


Figure 8.5: Participants often moved obstructing objects using the Pie Menu (under condition 2).

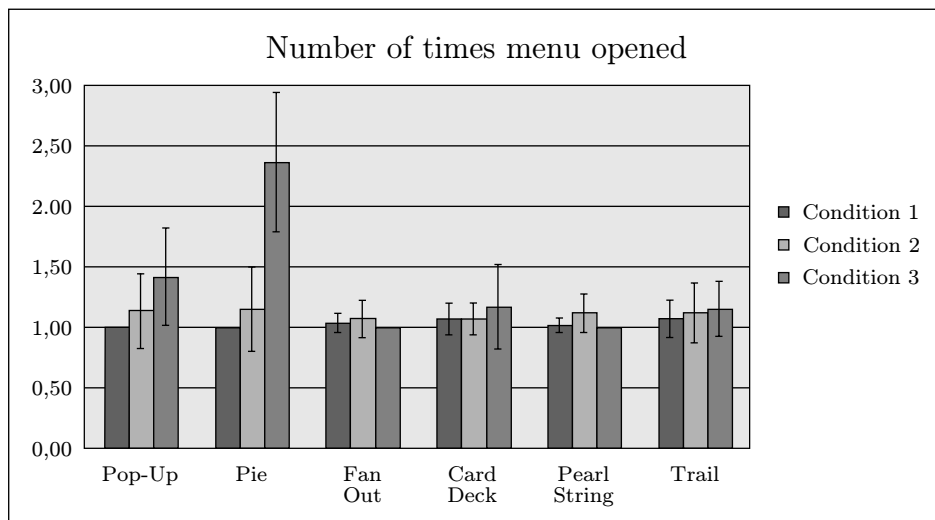


Figure 8.6: How often did participants open the menu to find an adequate position for an optimal item selection?

menu position was shorter than finding an adequate menu position (cf. Figure 8.7). A repeated measured ANOVA showed that there were significant differences between the menus under the three conditions (Condition 1: $F_{11,60} = 11,69, p < 0,001$; Condition 2: $F_{11,60} = 13,099, p < 0,001$; Condition 3: $F_{11,60} = 7,5, p < 0,001$).

These lend support to hypothesis H1 - hence, both the conventional

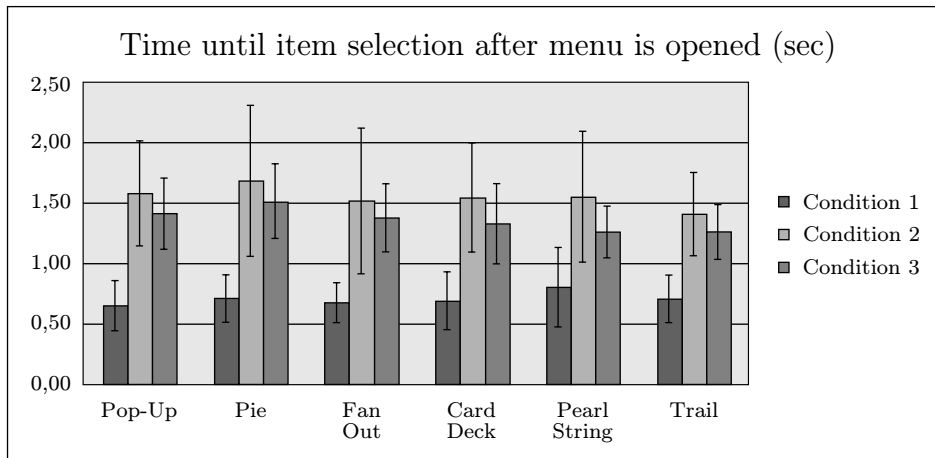


Figure 8.7: Selecting an item after having positioned the menu takes less time than finding an adequate menu position.

Pop-Up Menu and the Pie Menu demonstrated a better performance under condition 1. Moreover the results also support hypothesis H2 that user-drawn path menus were faster on a cluttered table.

Error Detection Analysis

Overall, there were little differences between the menus in terms of selecting the wrong item ($F_{11,60} = 1,16, p = 0,32$) with a mean error rate of 5% for the Pie Menu and 0 - 1.7% for the user-drawn menu (all under condition 2 and 3). The Trail Menu was the worst with a mean error rate of 3.3% on the empty table. We observed users moving the Trail Menu unintentionally before making a selection, which could explain the error rate.

Subjective Preference Analysis

In the post-study questionnaire, participants were asked to assign an overall rank to each of the six different menus and a preferred menu type for each of the three table conditions. In the 12 following questions, the participants rated the ease of use of each menu type for empty and cluttered tables, using a 5-point Likert-scale (1 = totally disagree, 5 = totally agree).

All participants were asked, which menu style they preferred for the empty table. 6 users chose the rectangular Pop-Up Menu as their favorite, 5 users chose the Pie Menu as their favorite, and 1 participant preferred the user-drawn menu on the empty table. On the obscured table with movable elements, none of the users reported the Pie Menu as a favorite. 2 users still preferred the Pop-Up Menu under this condition, but noted that they would prefer the user-drawn menu for menus containing more than 8 items.

Table 8.1: Subjective Survey Responses.

Menu	Empty Table	Cluttered Table
Pop-Up Menu	4.92 (SD 0.29)	2.29 (SD 1.24)
Pie Menu	4.75 (SD 0.45)	1.75 (SD 1.22)
Fan Out Menu	4.00 (SD 0.74)	3.83 (SD 0.94)
Card Deck Menu	4.25 (SD 0.87)	3.83 (SD 0.72)
Pearl String Menu	4.25 (SD 0.67)	4.25 (SD 0.75)
Trail Menu	4.50 (SD 0.67)	4.33 (SD 0.98)

All participants chose the user-drawn menu as their favorites under the 3rd condition, where they had no possibility to move the occluding elements.

Table 8.1 shows the results of the question about how easy the users felt it was to use each menu under two different conditions (empty table, cluttered table).

The overall ranking showed strong preferences for the Trail Menu. Participants, who ranked this metaphor highest, claimed the main reason was the possibility to move it around on a cluttered table until the optimal position and appearance was reached. All of the other menus had to be re-drawn or re-opened if they were not placed right the first time. One participant reported that the user-drawn Trail Menu seems to be “handy” for cluttered table because of the possibility to move the whole menu even after all menu items were already displayed. From the user study, we also noticed that none of the participants had troubles to adapt to the user-drawn menus and the pen hardware interface.

One drawback users did not like was the fact that they did not see the menu length of the drawn menus immediately, which makes the decision of where to start drawing the menu more difficult. They suggested “visualizing” the menu length. Some users noted no fundamental differences between the different types of user-drawn path menus, but recognized them to be advantageous when used on the cluttered table. And finally, one user suggested visualizing the link between the items of the Fan Out Menu.

8.2.5 Discussion

Performance

We were surprised to find out that the pMenus performed very well not only on cluttered tabletops but also on the empty table. The study revealed, that with an increasing menu size (i.e. pie menu), the performance of traditional context menus drops drastically on cluttered tabletops. However, the results

of the study only indicate a tendency, since the following parameters could influence the results of the study or change them completely:

- The simulated objects obscured approximately 66% of the screen. Depending on the utilization of the tabletop display, it could be covered with less or even more items.
- We used digital content instead of real objects to simulate a counter-balanced setup. Consequently, participants could move the content exclusively with the digital pen, which allowed only one-handed interaction. While users could move real physical objects with their second hand, they would possibly have to move items like a keyboard, a full coffee mug etc. which can require more time than moving digital content.
- Users were standing at the user study, which allowed them to reach regions of the screen further away faster than if seated.
- The context menus could be opened anywhere on the screen. However, in a number of applications users are required to open the context menu at the location of a specific virtual item they want to modify. Therefore, they either have to move the occluding physical object or the virtual item in order to see the context menu.
- Animal names were chosen as menu item entries in the study. We observed users guessing the content of partly covered menu item entries, which would not be possible with more complicated or similar sounding entries.
- The menu items appeared in random order, sorting them due to their importance would likely have an effect on the performance of the individual menus.
- All menus in the study contained 8 entries. Longer or shorter context menus will very likely influence the performance results. The current results indicate that a larger size alters menu performance on cluttered tabletops as the pie menu performs worse than all other menu types. For this reason, pMenus could also outperform rectangular pop-up menus when more than 8 entries are necessary on a cluttered tabletop.

Error Detection

As discussed in the performance analysis, we noticed study participants guessing the names of occluded menu entries. Therefore we expected more erroneous selections when using the pop-up menus than the pMenus. We were surprised by the small number of erroneous selections with the pop-up menus. A reason could be that the animal names were known to the test

participants, so they were able to determine the content of occluded entries fast and accurate by guessing occluded letters. While more complex entries could influence these results, under real-world conditions experienced users might remember even complex menu entries and guess partly occluded menu items with sufficient results. Therefore, further testing would be required to come to a satisfying conclusion.

User Preference

The results show that users do feel that user-drawn path menus (pMenus) are more comfortable to use and perform well when working on a cluttered digital tabletop. However, since all participants were exposed to the menus for the first time, it is hard to say how much the user preference was biased by the novelty factor of this interaction metaphor. A long-time study under real conditions would be necessary to see how training affects user preferences.

8.3 Informal Evaluation of the PenTable Setup

A prototype of the PenTable system was demonstrated to different audiences on occasions like open-day events and lab visits. Figure 8.8 depicts the tabletop setup used for these demonstrations. Approximately 30 visitors tried the system and commented very positively about the tabletop display and the example applications. We observed that the tracking quality of the pen input device is crucial for a satisfying interaction with the user-drawn GUI. When the quality dropped due to a Bluetooth lag or an insufficient pattern printout, users became frustrated trying to draw shapes or menus. However, all users commented very positively on the interaction metaphors and hardware interfaces of the system and said they could imagine working with it on a daily basis. Even in daylight settings, image brightness, resolution and contrast of the tabletop projection was perceived sufficient, but reflections on the glossy surface of the acrylic tabletop cover were irritating to some users.



Figure 8.8: The test setup at the *Upper Austria University of Applied Sciences* with a 40-inch digital tabletop display.

Chapter 9

Conclusion and Future Work

Overall System: This thesis described the design considerations and implementation of the PenTable setup, a multi-purpose workplace augmented with a digital tabletop. A number of example applications have been developed to demonstrate possible usage scenarios for the tabletop display as a peripheral display and additional PC input device. Initial feedback received from informal evaluations of the hardware setup and applications of the system was very positive. The current system functionality can be extended by programming custom tabletop applications. A user interface to connect generic tabletop sWidgets with Windows applications would allow users to create their own system extensions without programming experience.

Hardware Improvements: Touch input on the digital tabletop could be more convenient for using certain sWidgets than the pen. Therefore, a future version of the setup could combine Anoto tracking with systems like DiamondTouch [11] to provide both input possibilities. The current setup utilizes a top-mounted projector for the tabletop display. Future technology might permit the use of Electric Ink displays instead of projectors, which would reduce energy consumption and provide an improved viewing experience.

Tabletop GUI: The PenTable GUI features metaphors developed specifically for direct input tabletop displays. These metaphors include sketched widgets (sWidgets) instead of application windows and user-drawn path menus (pMenus) instead of menus. While previous research has introduced solutions for tabletop-specific GUI problems like multiple viewing angles and reach, the PenTable GUI is designed to avoid occlusions by physical objects on the tabletop. The results acquired in a first user study indicate a significant reduction of time when using the pMenus on obscured tabletop displays compared to standard pop-up menus. A future user study investigating performance with improved GUI functionality and changed tabletop scenarios

could provide further insights, as section 8.2.5 describes in more detail.

While the metaphors of the PenTable tabletop GUI are generic and can be applied to other tabletop systems, they still lack the complete feature set of a WIMP GUI. As a next step, cascading pMenus could be investigated to enable more complex menu structures. Current possibilities to modify the look of the sWidget applications are very limited. Future extensions could include tools to scale, rotate and modify color and shape of sWidgets after their creation. Text input with the *Microsoft Handwriting Recognition* has been investigated, but is currently not used in the system. The process of creating sWidgets is more flexible than most other sketched interface approaches, which try to determine a shape meaning and therefore limit user creativity. This possibility to create freeform GUI elements could be utilized to create a framework for interactive application interface prototypes similar to *Silk* by Landay [39].

Appendix A

Content of the CD-ROM

File System: Joliet

Mode: Single-Session

A.1 Thesis

Pfad: /

dm05015_leithinger_daniel_da.pdf Thesis (PDF-File)

A.2 Additonal Ressources

Pfad: /

images/ Images

bibliography/ Copies of websites (PDF-Files)

source/ PenTable source code

Consent Form

Title of the Study:

User-Drawn Menu Test

Researcher:

Daniel Leithinger

The aim of this study is to investigate the usability of different menu types on a tabletop setup. You will be introduced to the menus and then complete a number of tasks while the system records your actions and a researcher will observe you. You will later be asked to fill out a post-study questionnaire. The overall duration of the study will be approximately 20 minutes.

Confidentiality:

The anonymity of the collected data and the privacy of the subjects will be completely protected, and the information obtained from this data would be used only in theses, journal articles or conference publications written by the researchers. Only aggregate data will be reported in publications; the names and identities of the participants will not be published in any form.

Right to Withdraw:

Each participant is free to withdraw from the study at any time; this will not affect the participant's academic status. In the event of a participant withdrawing from the study, his / her data will be deleted. The participants will be advised of any new information that will have a bearing on the participants' decision to continue in the study.

Questions:

If you have any questions concerning the study, please feel free to ask at any point; you are also free to contact the researchers at the numbers provided above if you have questions at a later time.

If you wish to acquire information on the results of the research once the study is completed, send a request to Daniel Leithinger at daniel.leithinger@fh-hagenberg.at.

Consent to Participate:

I have read and understood the description provided above; I have been provided with an opportunity to ask questions and my questions have been answered satisfactorily. I consent to participate in the study described above, understanding that I may withdraw this consent at any time.

Name: _____

Date: _____ Signature: _____

Signature of the Researcher: _____

Post-Study Questionnaire

1. **Gender:** Male Female

2: **Age:** 18 – 24 25 – 34 35 – 44 45 +

3: **Highest Level of Education Completed/ Enrolled in?**

High School Bachelor Master Post-Graduate

4: **Handedness:** Right Left

5: **Have you ever used a Tabletop System before?** Yes No

6: **Have you ever used a pen-based computer input device before (Graphics Tablet/ Tablet PC/ PDA/...)?** Yes No

7: **Did you ever work with a pen-based input device over a longer period of time?**

No Yes, 1 to 20 hours in total Yes, over 20 hours in total

8: **How much time do you spend on a computer a day?**

1-2 hours 3-5 hours 5+ hours

9: **Today you were exposed to a number of different context menus.**

Please rank them from 1 (Favourite) to 6 (Least Favourite) according to your overall impression:

___ Card Deck (a)

___ Fan Out (b)

___ Pearl String (c)

___ Trail (d)

___ Rectangular (e)

___ Pie (f)

10: **Which menu did you prefer for the empty table?** Letter: ___

11: **Which menu did you prefer for the table with movable elements?** Letter: ___

12: **Which menu did you prefer for the table with fixed elements?** Letter: ___

Please rate the following statements by checking a box below them.

13: The Card Deck menu (a) was easy to use on the empty table.

Difficult Somewhat Difficult Neither Easy/ Difficult Somewhat Easy Easy

14: The Card Deck menu (a) was easy to use on the cluttered table.

Difficult Somewhat Difficult Neither Easy/ Difficult Somewhat Easy Easy

15: The Fan Out menu (b) was easy to use on the empty table.

Difficult Somewhat Difficult Neither Easy/ Difficult Somewhat Easy Easy

16: The Fan Out menu (b) was easy to use on the cluttered table.

Difficult Somewhat Difficult Neither Easy/ Difficult Somewhat Easy Easy

17: The Pearl String menu (c) was easy to use on the empty table.

Difficult Somewhat Difficult Neither Easy/ Difficult Somewhat Easy Easy

18: The Pearl String menu (c) was easy to use on the cluttered table.

Difficult Somewhat Difficult Neither Easy/ Difficult Somewhat Easy Easy

19: The Trail menu (d) was easy to use on the empty table.

Difficult Somewhat Difficult Neither Easy/ Difficult Somewhat Easy Easy

20: The Trail menu (d) was easy to use on the cluttered table.

Difficult Somewhat Difficult Neither Easy/ Difficult Somewhat Easy Easy

21: The Rectangular menu (e) was easy to use on the empty table.

Difficult Somewhat Difficult Neither Easy/ Difficult Somewhat Easy Easy

22: The Rectangular menu (e) was easy to use on the cluttered table.

Difficult Somewhat Difficult Neither Easy/ Difficult Somewhat Easy Easy

23: The Pie menu (f) was easy to use on the empty table.

Difficult Somewhat Difficult Neither Easy/ Difficult Somewhat Easy Easy

24: The Pie menu (f) was easy to use on the cluttered table.

Difficult Somewhat Difficult Neither Easy/ Difficult Somewhat Easy Easy

Verbal Questionnaire

25. Could you imagine using a tabletop system on a regular basis?

Yes No

If yes, what would you use it for?

26. What did you like best about the menu type you rated best?

27. What did you like not about the menu type you rated worst?

28. Do you have any suggestions for ways to improve the software?

29. Please tell us any final thoughts and/or impressions you may have about the software of your experiences in this study.

30. Please tell us if there is anything you would change or add to this form.

Thank you very much for participating in this study!

Bibliography

- [1] ACCOT, J. and S. ZHAI: *More than dotting the i's — foundations for crossing-based interfaces*. In *CHI '02: Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 73–80, New York, NY, USA, 2002. ACM Press.
- [2] AGARAWALA, A. and R. BALAKRISHNAN: *Keepin' it real: pushing the desktop metaphor with physics, piles and the pen*. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pp. 1283–1292, New York, NY, USA, 2006. ACM Press.
- [3] ALIAKSEYEU, D., J.-B. MARTENS and M. RAUTERBERG: *A computer support tool for the early stages of architectural design*. *Interacting with Computers*, 18(4):528–555, 2006.
- [4] APITZ, G. and F. GUIMBRETIERE: *CrossY: a crossing-based drawing application*. In *UIST '04: Proceedings of the 17th annual ACM symposium on User interface software and technology*, pp. 3–12, New York, NY, USA, 2004. ACM Press.
- [5] ARTMAN, H., R. RAMBERG, H. SUNDHOLM and T. CERRATTO-PARGMAN: *Action context and target context representations: a case study on collaborative design learning*. In *CSCL '05: Proceedings of the 2005 conference on Computer support for collaborative learning*, pp. 1–7. International Society of the Learning Sciences, 2005.
- [6] BAUDISCH, P., N. GOOD, V. BELLOTTI and P. SCHRAEDLEY: *Keeping things in context: a comparative evaluation of focus plus context screens, overviews, and zooming*. In *CHI '02: Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 259–266, New York, NY, USA, 2002. ACM Press.
- [7] BUSH, V.: *As We May Think..* *The Atlantic Monthly*, 176(1):101–108, July 1945.
- [8] COTTING, D. and M. GROSS: *Interactive environment-aware display bubbles*. In *UIST '06: Proceedings of the 19th annual ACM symposium*

- on *User interface software and technology*, pp. 245–254, New York, NY, USA, 2006. ACM Press.
- [9] DAM, A. VAN: *Post-WIMP user interfaces*. *Commun. ACM*, 40(2):63–67, 1997.
- [10] DAVIS, M. R. and T. O. ELLIS: *The RAND Tablet: A Man-Machine Graphical Communication Device*. In *AFIPS Conference Proceedings (FJCC)*, vol. 26, pp. 325–331. Spartan Books, 1964.
- [11] DIETZ, P. and D. LEIGH: *DiamondTouch: a multi-user touch technology*. In *UIST '01: Proceedings of the 14th annual ACM symposium on User interface software and technology*, pp. 219–226, New York, NY, USA, 2001. ACM Press.
- [12] E INK CORPORATION: *E Ink Shows Latest Products And Research Advances*. URL, <http://www.eink.com/press/releases/pr101.html>, May 2007. Copy on CD-ROM (eink.pdf).
- [13] ELLIOTT, A. and M. A. HEARST: *A comparison of the affordances of a digital desk and tablet for architectural image tasks*. *International Journal of Human-Computer Studies*, 56(2):173–197, 2002.
- [14] ENGELBART, D.: *The augmented knowledge workshop*. In *Proceedings of the ACM Conference on The history of personal workstations*, pp. 73–83, New York, NY, USA, 1986. ACM Press.
- [15] ENGLISH, W. K., D. C. ENGELBART and M. L. BERMAN: *Display-Selection Techniques for Text Manipulation*. In *IEEE Transactions on Human Factors in Electronics*, vol. HFE-8, pp. 5–15, March 1967.
- [16] FINDLATER, L. and J. MCGRENERE: *A comparison of static, adaptive, and adaptable menus*. In *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 89–96, New York, NY, USA, 2004. ACM Press.
- [17] FITTS, P. M.: *The information capacity of the human motor system in controlling the amplitude of movement..* *Journal of Experimental Psychology*, 47(6):381–391, June 1954.
- [18] FITZMAURICE, G., A. KHAN, R. PIEKÉ, B. BUXTON and G. KURTENBACH: *Tracking menus*. In *UIST '03: Proceedings of the 16th annual ACM symposium on User interface software and technology*, pp. 71–79, New York, NY, USA, 2003. ACM Press.
- [19] FORLINES, C., D. WIGDOR, C. SHEN and R. BALAKRISHNAN: *Direct-touch vs. mouse input for tabletop displays*. In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 647–656, New York, NY, USA, 2007. ACM Press.

- [20] GENTNER, D. and J. NIELSEN: *The Anti-Mac interface*. Communications of the ACM, 39(8):70–82, 1996.
- [21] GUIMBRETIERE, F. and T. WINOGRAD: *FlowMenu: combining command, text, and data entry*. In *UIST '00: Proceedings of the 13th annual ACM symposium on User interface software and technology*, pp. 213–216, New York, NY, USA, 2000. ACM Press.
- [22] HAIGH, T.: *Remembering the Office of the Future: The Origins of Word Processing and Office Automation*. IEEE Ann. Hist. Comput., 28(4):6–31, 2006.
- [23] HALLER, M., M. BILLINGHURST, D. LEITHINGER, J. LEITNER and T. SEIFRIED: *Coeno: enhancing face-to-face collaboration*. In *ICAT '05: Proceedings of the 2005 international conference on Augmented tele-existence*, pp. 40–47, New York, NY, USA, 2005. ACM Press.
- [24] HALLER, M., D. LEITHINGER, J. LEITNER, T. SEIFRIED, P. BRANDL, J. ZAUNER and M. BILLINGHURST: *The shared design space*. In *SIGGRAPH '06: Material presented at the ACM SIGGRAPH 2006 conference*, p. 29, New York, NY, USA, 2006. ACM Press.
- [25] HAN, J. Y.: *Low-cost multi-touch sensing through frustrated total internal reflection*. In *UIST '05: Proceedings of the 18th annual ACM symposium on User interface software and technology*, pp. 115–118, New York, NY, USA, 2005. ACM Press.
- [26] HANCOCK, M. S. and K. S. BOOTH: *Improving menu placement strategies for pen input*. In *GI '04: Proceedings of Graphics Interface 2004*, pp. 221–230, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2004. Canadian Human-Computer Communications Society.
- [27] HICK, W. E.: *On the rate of gain of information*. Quarterly Journal of Experimental Psychology, 4:11–36, 1952.
- [28] HINRICH, U., S. CARPENDALE and S. D. SCOTT: *Evaluating the effects of fluid interface components on tabletop collaboration*. In *AVI '06: Proceedings of the working conference on Advanced visual interfaces*, pp. 27–34, New York, NY, USA, 2006. ACM Press.
- [29] HOLLEMANS, G., S. VAN DE WIJDEVEN, T. BERGMAN and E. VAN LOENEN: *Entertaible: The Best of two Gaming Worlds*. MST News international newsletter on micro-nano integration, special issue on Fun and Recreation with Microsystems, pp. 9–12, 2006.

- [30] HURST, J., M. S. MAHONEY, N. H. TAYLOR, D. T. ROSS and R. M. FANO: *Retrospectives I: the early years in computer graphics at MIT, Lincoln Lab, and Harvard*. In *SIGGRAPH '89: ACM SIGGRAPH 89 Panel Proceedings*, pp. 19–38, New York, NY, USA, 1989. ACM Press.
- [31] HYMAN, R.: *Stimulus information as a determinant of reaction time*. *Quarterly Journal of Experimental Psychology*, 45:188–196, 1953.
- [32] JOHNSON, J., T. L. ROBERTS, W. VERPLANK, D. C. SMITH, C. H. IRBY, M. BEARD and K. MACKEY: *The Xerox Star: A Retrospective*. *Computer*, 22(9):11–26, 28–29, 1989.
- [33] JORDÀ, S., G. GEIGER, M. ALONSO and M. KALTENBRUNNER: *The reacTable: exploring the synergy between live music performance and tabletop tangible interfaces*. In *TEI '07: Proceedings of the 1st international conference on Tangible and embedded interaction*, pp. 139–146, New York, NY, USA, 2007. ACM Press.
- [34] KARAHALIOS, K. and T. BERGSTROM: *Visualizing Audio in Group Table Conversation*. In *TABLETOP '06: Proceedings of the First IEEE International Workshop on Horizontal Interactive Human-Computer Systems*, pp. 131–134, Washington, DC, USA, 2006. IEEE Computer Society.
- [35] KOIKE, H., Y. SATO and Y. KOBAYASHI: *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)*, 8(4):307–322, 2001.
- [36] KRUEGER, M. W., T. GIONFRIDDO and K. HINRICHSEN: *VIDEO-PLACE - an artificial reality*. In *CHI '85: Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 35–40, New York, NY, USA, 1985. ACM Press.
- [37] KURATA, T., T. OYABU, N. SAKATA, M. KOUROGI and H. KUZUOKA: *Tangible Tabletop Interface for an Expert to Collaborate with Remote Field Workers*. In *International Conference on Collaboration Technologies (CollabTech2005)*, pp. 58–63, 2005.
- [38] LAI, J., A. LEVAS, P. CHOU, C. PINHANEZ and M. VIVEROS: *BlueSpace: personalizing workspace through awareness and adaptability*. *International Journal of Human-Computer Studies*, 57(5):415–428, 2002.
- [39] LANDAY, J. A.: *SILK: sketching interfaces like crazy*. In *CHI '96: Conference companion on Human factors in computing systems*, pp. 398–399, New York, NY, USA, 1996. ACM Press.

- [40] LI, Y., K. HINCKLEY, Z. GUAN and J. A. LANDAY: *Experimental analysis of mode switching techniques in pen-based user interfaces*. In *CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 461–470, New York, NY, USA, 2005. ACM Press.
- [41] MACINTYRE, B., E. D. MYNATT, S. VOIDA, K. M. HANSEN, J. TULLIO and G. M. CORSO: *Support for multitasking and background awareness using interactive peripheral displays*. In *UIST '01: Proceedings of the 14th annual ACM symposium on User interface software and technology*, pp. 41–50, New York, NY, USA, 2001. ACM Press.
- [42] MCGUFFIN, M. J., L. TANCAU and R. BALAKRISHNAN: *Using Deformations for Browsing Volumetric Data*. In *VIS '03: Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*, pp. 401–408, Washington, DC, USA, 2003. IEEE Computer Society.
- [43] MICROSOFT CORPORATION: *After Two Decades, PC Use Is Still Rising in Businesses and Homes*. URL, <http://www.microsoft.com/presspass/press/2001/Aug01/08-0820YearPR.msp>, Aug. 2001. Copy on CD-ROM (PCsurvey.pdf).
- [44] MICROSOFT CORPORATION: *Microsoft Launches New Product Category: Surface Computing Comes to Life in Restaurants, Hotels, Retail Locations and Casino Resorts*. URL, <http://www.microsoft.com/presspass/press/2007/may07/05-29MSSurfacePR.msp>, May 2007. Copy on CD-ROM (surface.pdf).
- [45] MOGGRIDGE, B.: *Designing Interactions*. The MIT Press, October 2006.
- [46] MORGAN, R., J. WALNY, H. KOLENDA, E. GINEZ and F. MAURER: *Using Horizontal Displays for Distributed and Collocated Agile Planning*. In CONCAS, G., E. DAMIANI, M. SCOTTO and G. SUCCI (eds.): *XP*, vol. 4536 of *Lecture Notes in Computer Science*, pp. 38–45. Springer, 2007.
- [47] MORRIS, M. R.: *Benefits and Challenges of Tabletop Peripheral Displays*. In *Proceedings of UbiComp 2004 Extended Abstracts*, 2004.
- [48] NIELSEN, F.: *Visual Computing: Geometry, Graphics, and Vision*. Charles River Media / Thomson Delmar Learning, 2005.
- [49] PARKER, J. K., R. L. MANDRYK and K. M. INKPEN: *Integrating Point and Touch for Interaction with Digital Tabletop Displays*. *IEEE Computer Graphics and Applications*, 26(5):28–35, 2006.

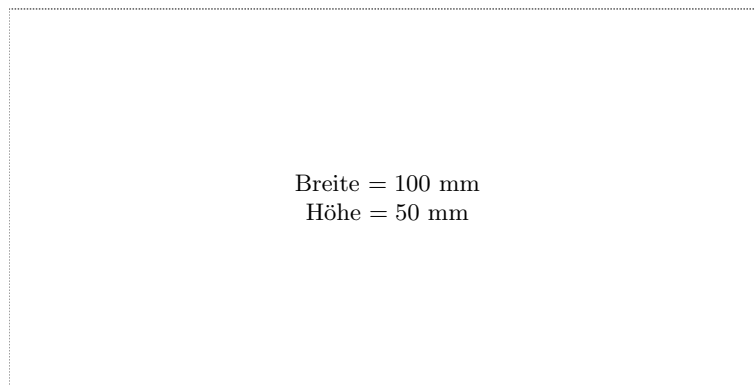
- [50] PATTEN, J., H. ISHII, J. HINES and G. PANGARO: *Sensetable: a wireless object tracking platform for tangible user interfaces*. In *CHI '01: Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 253–260, New York, NY, USA, 2001. ACM Press.
- [51] PATTEN, J., B. RECHT and H. ISHII: *Audiopad: a tag-based interface for musical performance*. In *NIME '02: Proceedings of the 2002 conference on New interfaces for musical expression*, pp. 1–6, Singapore, Singapore, 2002. National University of Singapore.
- [52] PINHANEZ, C.: *Using a steerable projector and a camera to transform surfaces into interactive displays*. In *CHI '01: CHI '01 extended abstracts on Human factors in computing systems*, pp. 369–370, New York, NY, USA, 2001. ACM Press.
- [53] PRANTE, T., N. STREITZ and P. TANDLER: *Roomware: Computers Disappear and Interaction Evolves*. *Computer*, 37(12):47–54, 2004.
- [54] REKIMOTO, J. and M. SAITOH: *Augmented surfaces: a spatially continuous work space for hybrid computing environments*. In *CHI '99: Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 378–385, New York, NY, USA, 1999. ACM Press.
- [55] ROBERTSON, G., M. CZERWINSKI, P. BAUDISCH, B. MEYERS, D. ROBBINS, G. SMITH and D. TAN: *The Large-Display User Experience*. *IEEE Computer Graphics and Applications*, 25(4):44–51, 2005.
- [56] SCHMIDT, A., M. STROHBACH, K. V. LAERHOVEN and H.-W. GELLERSEN: *Ubiquitous Interaction - Using Surfaces in Everyday Environments as Pointing Devices*. In *User Interfaces for All*, pp. 263–279, 2002.
- [57] SCOTT, S. D. and S. CARPENDALE: *Guest Editors' Introduction: Interacting with Digital Tabletops*. *IEEE Computer Graphics and Applications*, 26(5):24–27, 2006.
- [58] SEARS, A. and B. SHNEIDERMAN: *Split menus: effectively using selection frequency to organize menus*. *ACM Trans. Comput.-Hum. Interact.*, 1(1):27–51, 1994.
- [59] SELLEN, A. J. and R. H. HARPER: *The Myth of the Paperless Office*. MIT Press, Cambridge, MA, USA, 2003.
- [60] SHEN, C., N. LESH and F. VERNIER: *Personal digital historian: story sharing around the table*. *interactions*, 10(2):15–22, 2003.
- [61] SHEN, C., K. RYALL, C. FORLINES, A. ESENTER, F. D. VERNIER, K. EVERITT, M. WU, D. WIGDOR, M. R. MORRIS, M. HANCOCK

- and E. TSE: *Informing the Design of Direct-Touch Tabletops*. IEEE Computer Graphics and Applications, 26(5):36–46, 2006.
- [62] SIMPSON, R., A. RENEAR, E. MYLONAS and A. VAN DAM: *50 years after “As we may think”: the Brown/MIT Vannevar Bush symposium*. interactions, 3(2):47–67, 1996.
- [63] SMART TECHNOLOGIES INC.: *SMART Board for Plasma Displays - PP250 for 50” Pioneer*, 2001. <http://smarttech.de/products/shared/pdf/pp250.pdf>, Copy on CD-ROM (pp250.pdf).
- [64] SONG, H., F. GUIMBRETIERE, C. HU and H. LIPSON: *ModelCraft: capturing freehand annotations and edits on physical 3D models*. In *UIST ’06: Proceedings of the 19th annual ACM symposium on User interface software and technology*, pp. 13–22, New York, NY, USA, 2006. ACM Press.
- [65] SUTHERLAND, I. E.: *Sketchpad: A man-machine graphical communication system (Outstanding dissertations in the computer sciences)*. Garland Publishing, Inc., New York, NY, USA, 1980.
- [66] TANDLER, P., N. STREITZ and T. PRANTE: *Roomware-Moving Toward Ubiquitous Computers*. IEEE Micro, 22(6):36–47, 2002.
- [67] TSE, E., C. SHEN, S. GREENBERG and C. FORLINES: *Enabling interaction with single user applications through speech and gestures on a multi-user tabletop*. In *AVI ’06: Proceedings of the working conference on Advanced visual interfaces*, pp. 336–343, New York, NY, USA, 2006. ACM Press.
- [68] UNDERKOFFLER, J. and H. ISHII: *Urp: a luminous-tangible workbench for urban planning and design*. In *CHI ’99: Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 386–393, New York, NY, USA, 1999. ACM Press.
- [69] WEISER, M.: *The computer for the 21st century*. ACM SIGMOBILE Mobile Computer and Communication Review, 3(3):3–11, 1999.
- [70] WELLNER, P. D.: *Interacting with Paper on the DigitalDesk*. Master’s thesis, University of Cambridge Computer Laboratory, Cambridge, England, October 1993.
- [71] WILSON, A. D.: *PlayAnywhere: a compact interactive tabletop projection-vision system*. In *UIST ’05: Proceedings of the 18th annual ACM symposium on User interface software and technology*, pp. 83–92, New York, NY, USA, 2005. ACM Press.

- [72] WISNESKI, C., H. ISHII, A. DAHLEY, M. G. GORBET, S. BRAVE, B. ULLMER and P. YARIN: *Ambient Displays: Turning Architectural Space into an Interface between People and Digital Information*. In *CoBuild '98: Proceedings of the First International Workshop on Co-operative Buildings, Integrating Information, Organization, and Architecture*, pp. 22–32, London, UK, 1998. Springer-Verlag.

Messbox zur Druckkontrolle

— Druckgröße kontrollieren! —



— Diese Seite nach dem Druck entfernen! —