Improving Menu Interaction for Cluttered Tabletop Setups with User-Drawn Path Menus

Daniel Leithinger, Michael Haller
Upper Austria University of Applied Sciences
Softwarepark 11, 4232 Hagenberg, Austria
{daniel.leithinger, michael.haller}@fh-hagenberg.at

Abstract

Many digital tabletop systems have a graphical user interface (GUI) that features context (or pop-up) menus. While linear and pie menus are commonly used for direct pen and touch interaction, their appearance can be problematic on a digital tabletop display, where physical objects might occlude menu items. We propose a user-drawn path menu that appears along a custom path to avoid such occlusions. This paper introduces four different metaphors for user-drawn context menus: the Fan Out Menu, the Card Deck Menu, the Pearl String Menu, and the Trail Menu. It also presents the results we acquired from a user study, where participants were able to work faster on cluttered tabletop setups when using our user-drawn menus.

1. Introduction

Since the introduction of Wellner's DigitalDesk in 1991 [18] numerous PC interfaces based on digital tabletop setups have emerged [16]. Several metaphors have been proposed to deal with the problem of direct interaction with information on large surfaces [1]. These metaphors, however, have been developed for displays only obscured by the user's hand, which means applying them to digital tabletops would require an empty desk. Figure 1 depicts a table in a meeting room to illustrate the problem. Assuming this table is equipped with a digital tabletop and with a standard GUI, the work area would have to be emptied prior to each use. However, in a number of scenarios, meeting participants might need to use the physical objects and the digital tabletop simultaneously. When the user works on the tabletop next to a physical object and opens a context menu, its size, layout and position can cause it to overlap with objects nearby. In such a case, either the object or the menu has to be moved. Either one of these options disrupts the workflow and possibly leads to a suboptimal workspace arrangement.

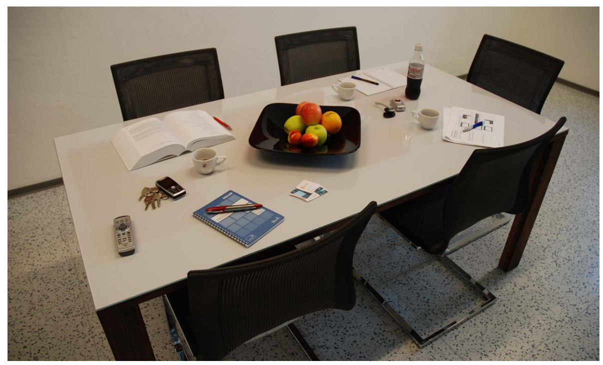

Figure 1: A meeting room with typical objects like notebooks, pens, fruits, coffee cups and mobile phones on the table.

Figure 2 depicts a scenario where both a rectangular drop-down menu and a pie menu fail on a table setup, due to visual overlapping with physical objects. We propose a *user-drawn path menu* which allows a customizable arrangement of the menu items and is therefore easily adaptable to the available tabletop display space.

(a) (b) (c)

Figure 2: (a) drop-down menus and (b) pie menus become hard to see or even unusable when overlapping with physical objects. (c) We propose a user-drawn path menu, which allows an optimal placement of the menu.

In the next section, we present related research, followed by a closer description of our four user-drawn menus. Then we present a user study followed by a detailed discussion of the results. Finally, we conclude with an outlook of further research.

2. Related Work

2.1. Digital Tabletops

Wellner introduces the approach projecting content onto a tabletop for an intuitive computer interface [18]. DigitalDesk is designed to provide a seamless link between real paper and the PC through the augmented tabletop. Ishii and Ulmer's metaDESK [17] integrates physical objects on a digital tabletop as part of a tangible user interface. Rekimoto and Saitoh introduce Augmented Surfaces [14], which allow users to exchange data with their laptop computers on an interactive table. Objects on the tabletop are tracked with optical markers and that way part of the user interface. DiamondTouch [5] is a digital tabletop surface supporting multiple users. In contrast to other input devices, the tracking technology is not disturbed by physical objects on the table. Interactive Environment-Aware Display Bubbles [4] are designed to map rectangular content onto freeform shapes for a space efficient arrangement of content on the table surface. Physical objects on the table are tracked with a top-mounted camera to avoid collisions with interface bubbles. To select bubble operations, a pie menu is used.

2.2. Pen Input

In our test setup, users interact directly on the tabletop display with a digital pen as a stylus. The following research has investigated the problems raised by direct touch or pen input and proposed a number of solutions.

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]. Apitz and Guimbretière propose an alternative to the standard WIMP-interface for tablet PCs entirely based on crossing [3]. Examples of context menus developed especially for pen input include FlowMenu [8] and Tracking Menus [7]. Tracking Menus were designed to prevent round trips to tool palettes with the pen.

Since the occlusion shadow of a user's hands on a direct input display can obstruct context menu options, several strategies have been proposed to improve the placement of such menus [9]. These strategies determine the user's handedness to predict a possible occlusion and adapt the menu position accordingly. Li et al. investigate various mode switching techniques in pen-based user interfaces [12]. Their findings can be applied to context menu activation on the tabletop, which is not investigated in this paper.

Most menus for direct pen or touch input have been designed for either small displays (e.g. PDAs, tablet PCs) or for large non-cluttered surfaces. In contrast to the related work, user-drawn path menus are designed especially for avoiding occlusions by physical objects.

3. User-Drawn Path Menus

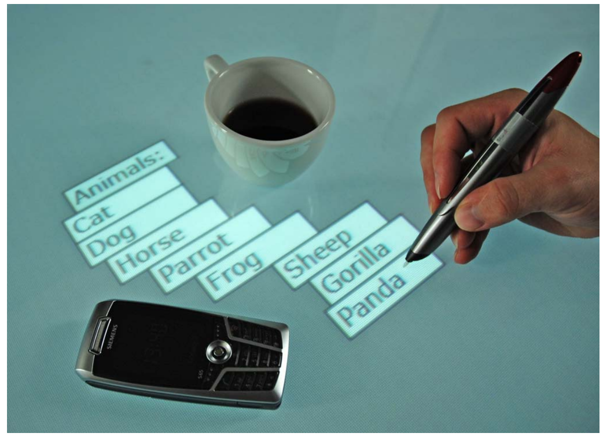

Figure 3: A user-drawn path menu.

Previous research to avoid menu occlusions produces very good results when dealing only with the user's hand [9]. But possible occlusion shadows created by other physical objects are substantially harder to determine. These occlusions can be even more problematic, since items like full coffee cups are more cumbersome to move than the user's hand.

To avoid possible menu occlusions, the system could try to determine the size and position of all physical objects on the tabletop with a camera. It could then adapt the context 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.

Our approach is different, since our system does not predict possible occlusions, but lets the user decide the menu size and shape. Instead of the context menu popping up after a user input, it appears along a user-drawn path (see Figure 3). This metaphor derives from the layers-as-a-stack-of-cards analogy [13], which is based on the idea of representing information layers as individual cards of a card deck. Each menu item is represented as a card and placed along the user-drawn path in different ways. This metaphor has the following advantages. Firstly, the user can easily create a menu not occluded by physical items by simply drawing a path around them. Secondly, since the menu placement and shape is created by the user, it poses less confusion than an automated placement. We propose four different metaphors to position the menu items along the path, which derive from natural ways of spreading information. In each of these metaphors, only the menus' root element appears when the input device first touches the surface. The menu items are placed along a path, created by the pen tip, and stay on their last placed position when the pen is lifted. The user can then select an item by tapping on it with the pen tip or close the menu by tapping on the root item.

3.1. Fan Out Menu

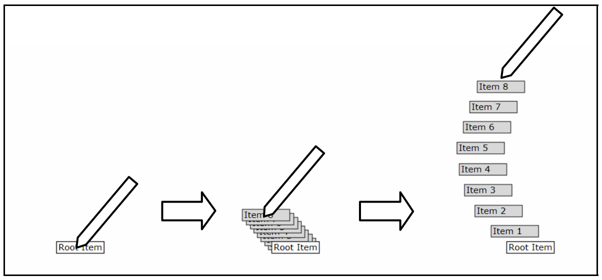

Figure 4: Fan Out Menu. The root item appears first then all menu items are spread along the path simultaneously.

This metaphor is similar to the approach described by Agarawala and Balakrishnan [2] to display information by fanning it out along a custom 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 (see Figure 4). The advantage of this metaphor is that since all menu items are spread out along the path simultaneously, the number of items is easy to determine while the stroke is drawn. Therefore the path length needed to display all menu items can be predicted by the user. The disadvantage to the other user-drawn menus is that it is always necessary to spread all items before selecting one.

3.2. Card Deck Menu

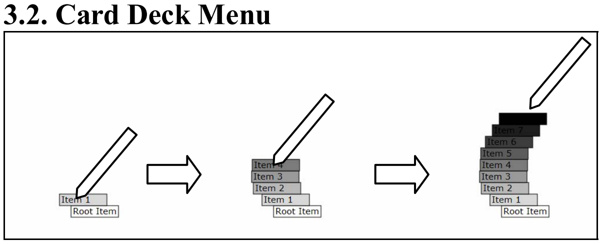

Figure 5: Card Deck Menu. Darker items appear later.

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 menus 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 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 (see Figure 5). An advantage of the card deck menu is the possibility to place a new menu item on top of any item other then 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.

3.3. Pearl String Menu

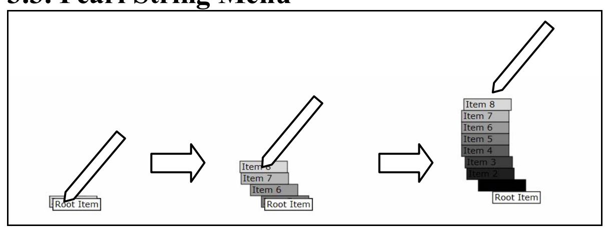

Figure 6: Pearl String Menu. Darker items appear later.

The behavior of the Pearl String Menu is very similar to the Card Deck Menu, but the order in which the menu items appear is reversed. When the pen tip touches the surface, the menus root element is displayed. While the pen moves, the first menu item follows the position of the pen tip. Each subsequent menu item is placed 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 (see Figure 6).

3.4. Trail Menu

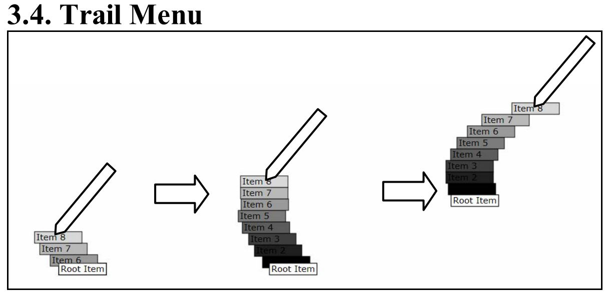

Figure 7: Placement of menu items in the Trail Menu. Darker items appear later.

The behavior of the Trail Menu is similar to the Pearl String Menu. But after all menu items are visible, they follow the pen tip like a tail instead of staying at their position (see Figure 7).

3.5. Implementation

All prototype menus were created using C# with Windows Presentation Foundation (WPF) for rendering. The test system either required Windows XP or Windows Vista.

4. Hypotheses

According to the Hick-Hyman Law [10][11] and to Fitts' Law [6], our proposed menus 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[1]. 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).

5. Experiment

5.1. Apparatus

We conducted a laboratory user study where users stand in front of a horizontal, rear-projected table (112 x 85 cm). The resolution of the projected display was 1024 x 768 pixels (cf. Figure 8).

The tracking of the stylus is realized by using a large Anoto pattern printed on a rear-projection foil in conjunction with digital pens (we used the Bluetooth digital pen from Maxell at 50 Hz). The pattern is clamped in-between two acryl panels to provide a stable and robust surface while protecting the pattern from scratches. The advantage of the Anoto tracking technology is a high tracking resolution, which is unaffected by physical objects on the tabletop. The tabletop was accompanied by a 19" LCD screen.

Figure 8: The rear-projection table and the LCD display used in our experiment.

5.2. 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).

5.3. 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 x 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.

[1] In the following, we use the term pop-up menu for all rectangular context menus and pie menu for a circular context menu.

5.4. Conditions

We used three different conditions for our experiment:

1. *Empty table:* users performed the task on an empty table (cf. Figure 8).
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 where 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 9).
3. *Obscured table with non-movable content:* in contrast to condition 2, the digital content could not be moved (cf. Figure 9).

Figure 9: The rear-projection table and the LCD display used in our experiment.

5.5. 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 participants x 5 trials x 6 menu types x 3 conditions = 1,890 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 errors.

5.6. Results

5.6.1. Performance Analysis

Figure 10 depicts the overall time participants used for the six menus under the three conditions[2]. 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 didn't find any difference between the user-drawn menus and pie menus (although users had to draw the path menu which usually should take longer than just a single pen click).

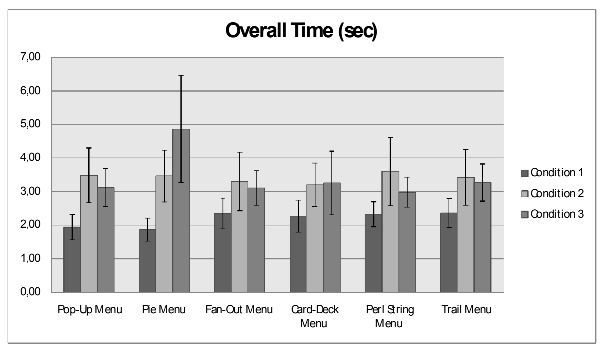

Figure 10: Overall time for each menu under the three different conditions.

We were surprised about the performance under condition 2, where the participants had the possibility to move the content. Although users had to move the digital content before getting the "traditional" menus under optimal conditions, they were not slower than sketching the path for the user-drawn menu. Under real conditions, however, users would have to move real printouts and obstacles, which is very much dependant on the type of object. Thus it can require more time than to move digital projected content.

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

[2] All spreadsheets are available at www.officeoftomorrow.org

(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 11). 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 condition 3 ($F_{11,60} = 5.83$, $p < 0.001$).

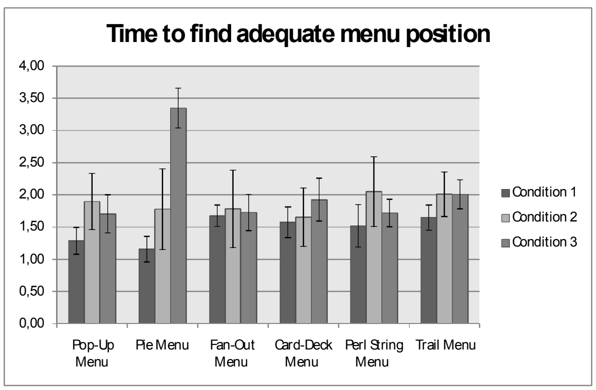

Figure 11: Time required to find an adequate position for the menu.

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 12 shows the average of moved objects under the second condition. Not surprisingly, 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 in the scenario, where he/she could do it.

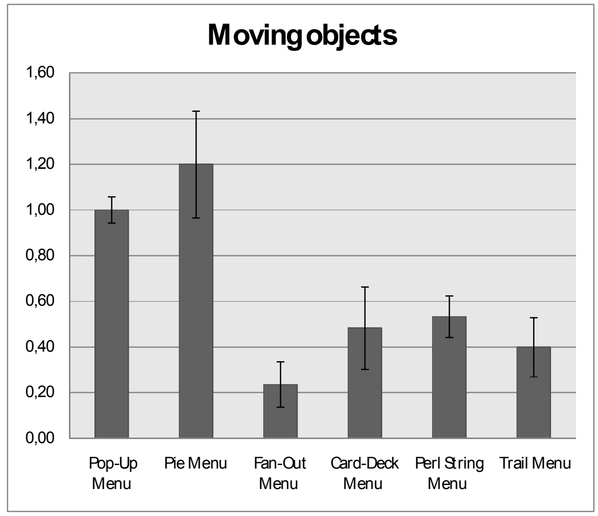

Figure 12: Participants often moved obstructing objects using the Pie Menu (under condition 2).

Figure 13 depicts how often participants had to open the menu to get the adequate position for selecting the target item.

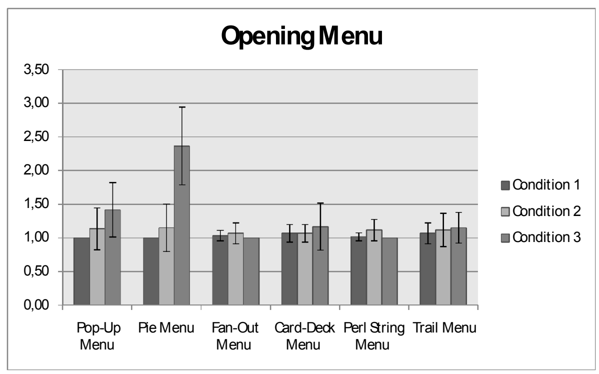

Figure 13: How often did participants open the menu to find an adequate position for an optimal item selection?

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 menu position was, of course shorter (cf. Figure 14). 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: $F_{11,60} = 7.5$, $p < 0.001$).

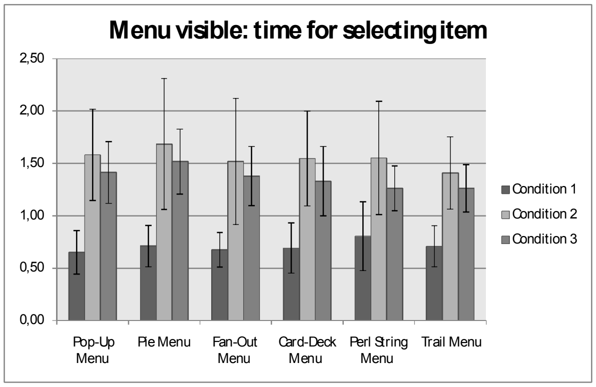

Figure 14: Selecting an item after having positioned the menu takes less time than finding an adequate menu position.

These lend support to hypothesis *H1* – hence, both the conventional Pop-Up Menu and the Pie Menu demonstrated a better performance under condition 1. Moreover our results also support hypothesis *H2*, where our proposed user-drawn path menus were faster on a cluttered table.

5.6.2. 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% once the table was empty. We observed users moving the Trail Menu unintentionally before making a selection, which could explain the error rate.

5.6.3. Subjective Preference Analysis

In the post-study questionnaire we asked the participants 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).

We asked all participants 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. 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 1 shows the results of the question about how easy the users felt it was to use the menu under two different conditions (empty table, cluttered table).

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)

Table 1: Subjective Survey Responses.

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 and he/she liked 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 new interface.

One drawback users didn't like was the fact that they didn't 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.

6. Discussion

The results show that users do feel that user-drawn menus make a strong sense while working on a cluttered table. Even on an empty table, we could only find a short time lag while using our approach. In our experiment, we used digital content instead of real printouts to simulate a counterbalanced setup. Consequently, users could move the content with the digital pen. In a real environment, however, users would have to move the keyboard, the coffee mug etc. which can require more time, although it allows two handed interaction.

7. Conclusion and Future Work

In this paper, we have presented a user-drawn interface for tabletop interaction. Our work builds on previous research (e.g. Pie Menus, Tabletop Setups) and introduces a novel context menu. Our main observation was that current graphical user interface metaphors were perceived to be too inflexible for a tabletop cluttered with objects. In this paper, we presented four different user-drawn menus. The results we achieved in our first user study present a significant reduction of time using these menus on an obscured tabletop display. Moreover, users also postulated the user-drawn menus as their favorites. Our ongoing work will continue to add and analyze cascading user-drawn menus.

Acknowledgement

This project is sponsored by the Austrian Science Fund FFG (FHplus, contract no. 811407) and voestalpine Informationstechnologie GmbH. The authors would like to express their gratitude to the anonymous reviewers and to Sean Castle for their useful comments, and to all the participants of the user study.

References

[1] Accot, J. and Zhai, S. 2002. More than dotting the i's --- foundations for crossing-based interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Changing Our World, Changing Ourselves* (Minneapolis, Minnesota, USA, April 20 - 25, 2002). CHI '02. ACM Press, New York, NY, 73-80.

[2] Agarawala, A. and Balakrishnan, R. 2006. Keepin' it real: pushing the desktop metaphor with physics, piles and the pen. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Montréal, Québec, Canada, April 22 - 27, 2006). R. Grinter, T. Rodden, P. Aoki, E. Cutrell, R. Jeffries, and G. Olson, Eds. CHI '06. ACM Press, New York, NY, 1283-1292.

[3] Apitz, G. and Guimbretière, F. 2004. CrossY: a crossing-based drawing application. In *Proceedings of the 17th Annual ACM Symposium on User interface Software and Technology* (Santa Fe, NM, USA, October 24 - 27, 2004). UIST '04. ACM Press, New York, NY, 3-12. Canada, 2005.

[4] Cotting, D. and Gross, M. 2006. Interactive environment-aware display bubbles. In *Proceedings of the 19th Annual ACM Symposium on User interface Software and Technology* (Montreux, Switzerland, October 15 - 18, 2006). UIST '06. ACM Press, New York, NY, 245-254.

[5] Dietz, P. and Leigh, D. 2001. DiamondTouch: a multi-user touch technology. In *Proceedings of the 14th Annual ACM Symposium on User interface Software and Technology* (Orlando, Florida, November 11 - 14, 2001). UIST '01. ACM Press, New York, NY, 219-226.

[6] Fitts, P. The Information Capacity of the Human Motor System in Controlling the Amplitude of Movement. J. Experimental Psych., 47. 1954. 381-391.

[7] Fitzmaurice, G., Khan, A., Pieké, R., Buxton, B., and Kurtenbach, G. 2003. Tracking menus. In *Proceedings of the 16th Annual ACM Symposium on User interface Software and Technology* (Vancouver, Canada, November 02 - 05, 2003). UIST '03. ACM Press, New York, NY, 71-79.

[8] Guimbretiére, F. and Winograd, T. 2000. FlowMenu: combining command, text, and data entry. In *Proceedings of the 13th Annual ACM Symposium on User interface Software and Technology* (San Diego, California, United States, November 06 - 08, 2000). UIST '00. ACM Press, New York, NY, 213-216.

[9] Hancock, M. S. and Booth, K. S.: Improving Menu Placement Strategies for Pen Input. In *Proceedings of the Graphics Interface 2004 Conference* (May 17-19, 2004, London, Ontario, Canada). Heidrich W. and Balakrishnan R., Canadian Human-Computer Communications Society, 221-230

[10] Hick, W. On the rate of gain of information. J. Experimental Psychology, 4. 1952. 11-36.

[11] Hyman, R. Stimulus information as a determinant of reaction time. J. Experimental Psych., 45. 1953. 188-196.

[12] Li, Y., Hinckley, K., Guan, Z., and Landay, J. A. 2005. Experimental analysis of mode switching techniques in pen-based user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Portland, Oregon, USA, April 02 - 07, 2005). CHI '05. ACM Press, New York, NY, 461-470.

[13] McGuffin, M. J., Tancau, L., and Balakrishnan, R. 2003. Using Deformations for Browsing Volumetric Data. In *Proceedings of the 14th IEEE Visualization 2003* (Vis'03) (October 22 - 24, 2003). IEEE Visualization. IEEE Computer Society, Washington, DC, 53.

[14] Rekimoto, J., and Saitoh, M. 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*, pages 378–385, New York, NY, USA, 1999. ACM Press.

[15] Robertson, G., Czerwinski, M., Baudisch, P., Meyers, B., Robbins, D., Smith, G., and Tan, D. The large-display user experience. *IEEE Comput. Graph. Appl.*, 25(4):44–51, 2005.

[16] Scott, S. D. Territoriality in collaborative tabletop workspaces. PhD thesis, Calgary, Alta., Canada,

[17] Ullmer, B. and Ishii, H. 1997. The metaDESK: models and prototypes for tangible user interfaces. In *Proceedings of the 10th Annual ACM Symposium on User interface Software and Technology* (Banff, Alberta, Canada, October 14 - 17, 1997). UIST '97. ACM Press, New York, NY, 223-232.

[18] Wellner, P. 1993. Interacting with paper on the DigitalDesk. *Commun. ACM* 36, 7 (Jul. 1993), 87-96.