Tilt Menu: Using the 3D Orientation Information of Pen Devices to Extend the Selection Capability of Pen-based User Interfaces

Feng Tian[1], Lishuang Xu[1], Hongan Wang[1, 2], Xiaolong Zhang[3], Yuanyuan Liu[1], Vidya Setlur[4], Guozhong Dai[1, 2]

[1] Intelligence Engineering Lab, Institute of Software, Chinese Academy of Sciences
[2] State Key Lab. of Computer Science
P.O.Box 8718, Beijing, China
{tf, xls, wha, lyy, dgz}@iel.iscas.ac.cn

[3] The Pennsylvania State University,
307D IST Building, University Park, PA16802,
lzhang@ist.psu.edu
[4] Nokia Research Palo Alto
975 Page Mill Road, Suite 200, Palo Alto,
vidya.setlur@nokia.com

ABSTRACT

We present a new technique called 'Tilt Menu' for better extending selection capabilities of pen-based interfaces. The Tilt Menu is implemented by using 3D orientation information of pen devices while performing selection tasks. The Tilt Menu has the potential to aid traditional one-handed techniques as it simultaneously generates the secondary input (e.g., a command or parameter selection) while drawing/interacting with a pen tip without having to use the second hand or another device. We conduct two experiments to explore the performance of the Tilt Menu. In the first experiment, we analyze the effect of parameters of the Tilt Menu, such as the menu size and orientation of the item, on its usability. Results of the first experiment suggest some design guidelines for the Tilt Menu. In the second experiment, the Tilt Menu is compared to two types of techniques while performing connect-the-dot tasks using freeform drawing mechanism. Results of the second experiment show that the Tilt Menu perform better in comparison to the Tool Palette, and is as good as the Toolglass.

Author Keywords

Tilt menu, pen, orientation, selection, pen-based UI

ACM Classification Keywords

H.5.2 [User Interfaces]: Graphical User Interfaces, Theory and methods, Interaction styles.

INTRODUCTION

The most popular pointing device in traditional GUI is a mouse that provides positional information. However, more recently, the pen has gained attention as a new kind of input modality given its compatibility with natural interface experiences. The pen can not only provide 2D information of the pen tip, but also other information such as writing pressure, 3D orientation, and 3D rotation. There is some research on techniques to incorporate this information with pen-based interaction [25] [10] [30].

In this paper, we present a new type of menu based on 3D orientation, called the Tilt Menu. As shown in Figure 1, by using the pen's 3D orientation information to perform selection tasks, the Tilt Menu supports selection without any pen tip movements. Therefore, as a one-handed technique, the Tilt Menu can smoothly merge command selection and direct manipulation in freeform drawing tasks.

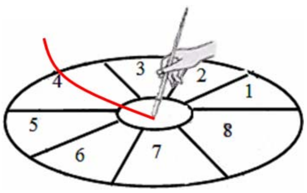

Figure 1. A right-handed user begins to perform a selection task with a Tilt Menu when freeform drawing.

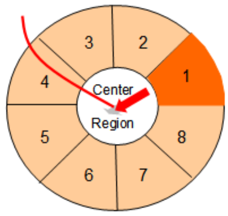

Figure 2. The menu's state corresponds to the user's action in Figure 1. Before the selection task is completed, the highlighted item gives the user a visual feedback of current pen-tilting direction.

As shown in Figure 2, the shape of the Tilt Menu is similar to that of a pie menu, consisting of several rounded, fan-shaped menu items [7]. However, unlike the pie menu, an item is not selected by the tap action of pen, but rather selected by tilting the pen as soon as the Tilt Cursor controlled by the pen, hits the edge of an item. The Tilt Cursor is a kind of cursor that dynamically reshapes itself to providing the 3D orientation cue of a pen [30].

The Tilt Menu can be useful in applications, such as freeform drawing and online synchronous sketch recognition. If the application requires freeform drawing, it is worthwhile to explore one-handed techniques that have the inherent advantage of merging command selection and direct manipulation [12]. However, current one-handed techniques all face similar limitations wherein the user has to move the pen tip in order to make a selection. Thus there is ambiguity regarding whether the selection stroke is part of the drawing or the drawing has to be started at some point away from the initial location where the technique was initiated. Comparing to other one-handed techniques, the Tilt Menu can smoothly merge command selection and freeform drawing simultaneously by using the pen's 3D orientation information to perform selection tasks.

Continuous interaction is a very important feature in pen-based user interfaces [14]. In an online synchronous sketch recognition system, the recognition procedure is simultaneously run with the drawing action [3] [29] [19], thus helping users know whether their intentions have been understood correctly or not. If the current recognition result is ambiguous, users can correct the recognition results via interaction in real time. Therefore, users have to perform some selection tasks simultaneously with sketching when the current recognition result is ambiguous. However, with the current techniques, such as SmartGuides [1] or automatic constrains generation [24], users have to cancel current drawing task to perform the selection. In such situations, the Tilt Menu can simultaneously generate the secondary input while sketching with a pen tip without having to use the second hand or another device.

Additionally, Tilt Menus can be used to enhance traditional context-aware menu techniques such as pie or marking menus [19] [32]. The 3D orientation information may be added to existing pen interaction techniques without interfering with interactions. By adding the 3D orientation information to the technique, the Tilt Menu makes the pen a kind of 3D interaction device.

RELATED WORK

Inking and gesturing are two principal modes in pen-based user interaction. Li investigates five techniques for switching between ink and gesture modes in pen interfaces, including a pen-pressure based mode switching technique that allows implicit mode transitions [20]. Saund presents an inferred-mode interaction protocol that ascertains the user's intent by using the stylus' trajectory and context [28]. While in some cases this protocol does not need an explicit command, the system presents a selector widget if the stroke drawn is ambiguous.

Selection-action patterns are widely used in pen interfaces. While the patterns are traditionally sequential, there have been efforts to improve this experience. Pigtail delimiters allow selection-action patterns to be performed in one continuous fluid stroke [13]. Here, a user explicitly creates a pigtail by intersecting his/her own stroke and then uses the stroke's direction to specify an action or manipulate an object. Pigtails provide a way to integrate an explicit command invocation in a fluid stroke following the selection specification. Pressure marks can encode selection-action patterns in a concurrent, parallel interaction [27]. Pen strokes where the variations in pressure make it possible to indicate both a selection and an action simultaneously. Zliding explores integrated panning and zooming by concurrently controlling input pressure while sliding in X-Y space [26].

Toolglass is one of the first interaction mechanisms to merge command selection and direct manipulation [6]. With Toolglass, the user uses his non-dominant hand to manipulate a translucent tool palette and his dominant hand to select commands and perform direct manipulation tasks. Bimanual marking menu is a two-handed technique that allows users to manipulate objects with one hand while issuing commands in parallel with the other [21]. FlowMenu is a radial menu that does not require users to lift the pen to select a command [11]. Rather, FlowMenu requires leaving and reentering the central rest area in specific directions to determine menu selection. Control menu is another radial menu that does not require users to lift the pen to select a command [23]. This menu uses a threshold distance as the triggering mechanism to determine menu selection. Guimbretière presents new evidence for the benefits of merging command selection and direct manipulation in commonly-performed direct manipulation tasks of modern interfaces indicating that current one-handed techniques cannot accomplish the merging in freeform drawing tasks [12]. This is the primary motivation for creating a novel pen-based technique, Tilt Menu.

Various localized menus have been designed, such as Marking Menu [18], Tracking Menu [8], and Hover Widgets [10]. However, most of them have to use the pen tip's movements in order to make a selection. The tilt information can be a new dimension to resolving this problem. The Rockin' Mouse which is a promising device for both 2D and 3D interaction that uses tilt input to integral 3D Manipulation on a Plane [4]. The Tilt Cursor is a type of cursor that dynamically reshapes itself to providing the 3D orientation cue of a pen [30]. Experimental results show that the Tilt Cursor can provide better stimulus-response compatibility on a touchpad when compared to other kinds of cursor. Some other work focuses on using tilt information of the phone to do text entry tasks for mobile devices, such as TiltType [22], TiltText [31].

THE TILT MENU

Design Properties

The Tilt Menu provides a number of beneficial properties:

New Command Layer

The Tilt Menu creates a new command layer by relying on the 3D orientation information of a pen. Traditional pen-based UI design uses pen-tip information, such as position, and pressure. Tilt Menus can be easily integrated into pen-based UI systems without conflicting with existing designs. Users do not need to worry about the interference of a Tilt Menu with ink and gestures created by the pen tip.

Integrated Data Entry and Data Manipulation

Data manipulation can be smoothly integrated with data entry as users do not need to lift the pen tip to invoke a menu. In particular, in scenarios like freeform drawing, online synchronous sketch recognition, users can benefit from merged data entry and data manipulation. For example, to change the color of a line in freeform drawing, the Tilt Menu allows users to perform a color-selection task by tilting the pen while the pen tip is still drawing the line. The Tilt Menu can generate the secondary input (e.g., a command or parameter selection) without using the second hand or another device when the pen tip is still involved in the primary task.

Localized UI

The center of a Tilt Menu is the position of the pen tip. Thus, the Tilt Menu is always local, saving a user time in locating a menu and reducing physical movements in selection. Meanwhile, a user tilts the pen to select a specific slice without moving the pen tip. Such convenience may further improve user performance.

Context-aware Menu Invoking

The Tilt Menu can be invoked based on the context of current task performed rather than an explicit command or user action. For example, in an online synchronous sketch recognition system, the Tilt Menu is automatically invoked when ambiguous recognition results appear during sketching. And the Tile Menu will automatically disappear when the context changes without cancelled by the user.

Design Space

Selection Design

We use 3D Cartesian coordinates to compute how the Tilt Cursor hits an item of the Tilt Menu, as shown in Figure 3.

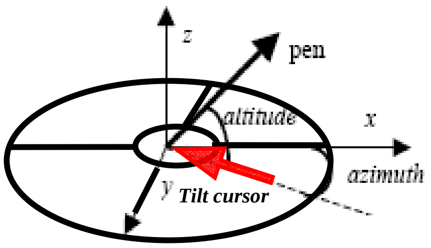

Figure 3. 3D coordinates for calculating the activation of an item in the Tilt Menu.

The head of the Tilt Cursor and the center of the Tilt Menu are always at the origin of the coordinates. The viewpoint is at the positive infinity of z axis above the cursor. The Tilt Cursor is a projection of the pen vector onto the XY-plane. The Tilt Cursor selecting an item of the Tilt Menu can be calculated according to the following steps:

(1) Computing the length of the Tilt Cursor: *TCLength*

$$\Delta x = \left(altAdjust - \frac{|altitude|}{altF} \right) \times \sin\left(\frac{azimuth}{aziF} \right)$$

$$\Delta y = \left(altAdjust - \frac{|altitude|}{altF} \right) \times \cos\left(\frac{azimuth}{aziF} \right)$$

$$TCLength = \sqrt{\Delta x^2 + \Delta y^2}$$

where Δx is the difference between the end and the head of the cursor in X axis; Δy is the difference between the end and the head of the cursor in Y axis; altAdjust is the altitude zero adjust; altF is the altitude factor; aziF is the azimuth factor; TCLength is the Tilt Cursor's length. [30]

(2) If *TCLength* is greater than the radius of the center region, then it means that an item is selected by the Tilt Cursor. We then calculate the azimuth value of the Tilt Cursor, to map the value to the actual item selected.

Interaction Transitions

Figure 4 shows a state transition diagram of the Tilt Menu. When the Tilt Menu is invoked (State 1), its center appears at the position of the pen tip. The Tilt Menu automatically follows the position of pen tip until a selection task is finished. With that, the 3D orientation information of the pen will determine the menu's next state. When the entire Tilt Cursor is in the center region of the Tilt Menu, the state becomes "selection starting" (State 3). In this state, users can use pen tilt to perform a selection task. Otherwise, the state will be "menu disable" (State 2). This means that before the selection, the Tilt Cursor has already hit the edge of an item. Users have to go to State 3 by changing the altitude value of the pen until the Tilt Cursor enters the interior of the center region. After that, an item will be selected (State 4) when the Tilt Cursor hits its edge.

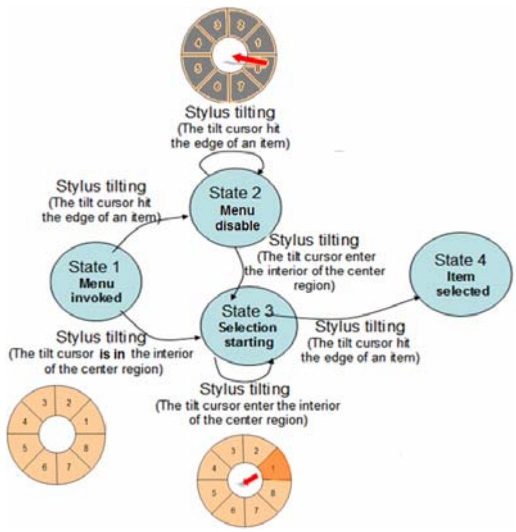

Figure 4. State transitions of the Tilt Menu

Visualization Techniques

To aid users in effectively using the Tilt Menu, we adopt a real time visual feedback of the pen and the menu state. We use the Tilt Cursor [30] to dynamically reflect the 3D visual cue of the pen device. As for the menu, the color of item is gray when the menu is in the state 2 “menu disable”. In the state 3 “selection starting”, the current highlighted item gives user a visual cue that the azimuth value of the pen device belongs to this item. If the user makes the azimuth value of pen stable, and keeps on tilting the altitude of the pen device, the highlighted item will be selected once the Tilt Cursor hits the item. After the selection is made, the menu disappears immediately. It should be noted that the Tilt Menu could have different color styles and the Tilt Cursor applied in the menu could have other appearances.

APPLICATION EXAMPLE

To better understand the Tilt Menu, we present an application of the Tilt Menu for an intelligent whiteboard system used in a high-school geometry class. One of the functions of the intelligent system is to estimate the intentions of users and then take appropriate actions. Consider a scenario where a teacher is drawing a circle within a given square. The circle could be within the square, be tangential to the square, or intersecting the square. If the system knows the intention of the teacher, an appropriate circle can be quickly drawn based on part of the circle the teacher has already drawn. Although the system can speculate three possible user intentions, the challenge here is how to let the teacher select and execute an action when the pen tip is still drawing the circle.

With the Tilt Menu, this problem can be easily addressed. When the teacher just draws part of the circle near to an edge of the square, a Tilt Menu can be presented around the cursor with menu items such as “separate”, “tangent”, and “intersect” corresponding to three possible outcomes. The teacher can then make a selection by tilting the pen towards a desired option and letting the system complete the circle (Figure 5).

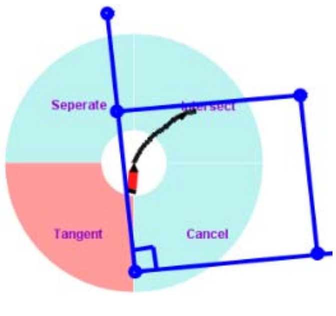

Figure 5. Drawing a circle within a given square

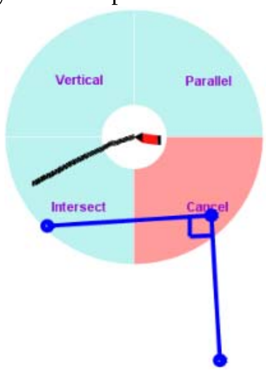

Figure 6. Drawing a right angle alongside of an existing right angle

The Tilt Menu is beneficial in two ways. From the user’s aspect, the menu allows the teacher to feed the system accurate intention information without having the on-going drawing task interrupted. From the system’s aspect, the drawing information, from the trajectory of the pen tip, and the user command, from the output of the Tilt Menu, can be easily distinguished. It should be noted that the Tilt Menu is not limited in use only to this case. For example, if the user wants to draw a right angle alongside of an existing right angle, the Tilt Menu can be shown two times to let the user select in order to make two parts of the stroke parallel with the corresponding existing line of the right angle, as shown in Figure 6.

EXPERIMENTS

Goals

To evaluate the usability performance of the Tilt Menu, we conduct two experiments. The goals of these experiments are to examine the following two issues:

- How do the parameters of the Tilt Menu, such as menu size (number of slices per menu) [18] and orientation of item affect the menu’s usability?
- How does the Tilt Menu with optimal parameters perform in comparison to the other menus in tasks concerning merging command selection and direct manipulation?

Experiment 1

This goal of this experiment is to study how the Tilt Menu’s parameters may affect usability. We use the touchpad to perform selection tasks with three types of Tilt Menus. These three types of Tilt Menus are different in menu size, including 4, 8 or 12 items. Subjects select target "slices" from the Tilt Menus as quickly and as accurately as possible. All Tilt Menus contain numbered segments, always with a "1" starting immediately on the positive (right) side of the x axis. The other slices are labeled in counter clockwise order with the maximum number immediately below the positive (right) side of the x axis as shown in Figure 7. The diameter of all Tilt Menus is 40 mm, and the diameter of the center region is 10 mm. We use Times New Roman 14 point bold font to label the items.

Figure 7. Numbered segments of Tilt Menus with 4, 8, 12 slices respectively.

Subjects and Apparatus

Twelve subjects, eight males and four females, participate in the experiment. Participants range in ages from 19 to 31. To minimize experimental bias due to handedness, we ensure that all participants are right-handed according to self-report. All participants have normal or corrected-to-normal vision. A Wacom 12’×12’ touchpad and a 17’ LCD screen with the resolution in 1024×768 pixels are used for the experiment.

Procedure

We use a completely within-subject experimental design. Participants are instructed to complete all trials in this experiment. A Latin square is used to balance the order in which participants use the Tilt Menu with three menu sizes. Each combination of trial parameters is repeated twelve times, thus a total of (4+8+12)×12 repetitions test trials are collected from each participant.

In each session (test trials for one menu size), trials are completely randomized in a manner consistent with other experiments of this kind [7] [18], meaning that every session of trials has a unique order of presentation and that every ordering is visibly different from all other orderings given to all participants. Participants are given 1 minute breaks between sessions.

Participants are given twenty-four practice warm-up trial sets to familiarize themselves with each menu size. Practice trials consist of trials presented in the same fashion as that of the experimental trials. The experiment totally lasts about 10 minutes for each participant.

Results

We first analyze how the menu size influences the performance of the Tilt Menu. As expected, increasing the menu size significantly increases response time ($F_{2,22}$ = 17.971, $p < .0001$), and error rates ($F_{2,22}$ = 79.920, $p < .0001$), as shown in Figure 8. The mean movement time of three menu types with 4, 8, 12 items are 794.57 ms, 933.55 ms and 1087.06 ms respectively. Pair-wise comparisons reveal significant differences between all menu types ($p < .04$). The mean error rate of the Tilt Menu with 4, 8, 12 items are 3.00%, 9.33% and 14.67% respectively. And pair-wise comparisons reveal significant differences between all menu types ($p < .0001$).

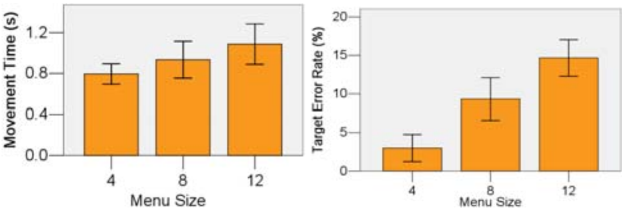

Figure 8. Comparison of the Tilt Menu's mean response time and mean error rate for three menu sizes

To further understand the impact of menu orientation on usability, we research the results of the Tilt Menu with 4, 8 and 12 items. As for the Tilt Menu with 4 items, repeated measures analysis of variance show a non-significant main effect for item orientations in response time ($F_{3,33}$ = .948, $p = .429$) and accuracy rate ($F_{3,33}$ = 1.693, $p = .187$). Results are shown in Figure 9. As for the response time, we observe that the items 2 and 4 are longer than that of items 1 and 3, but pair-wise comparisons reveal non-significant differences between all items ($p > .05$). Meanwhile, we observe that the error of items 2 and 4 are lower than that of other items 1 and 3. Pair-wise comparisons reveal non-significant differences between all items ($p > .05$). As for the Tilt Menu with 8 items, repeated measures analysis of variance show a significant main effect for item orientations in response time ($F_{7,77}$ = 8.226, $p < .0001$) and error rate ($F_{7,77}$ = 3.881, $p = .001$). Results are shown in Figure 10.

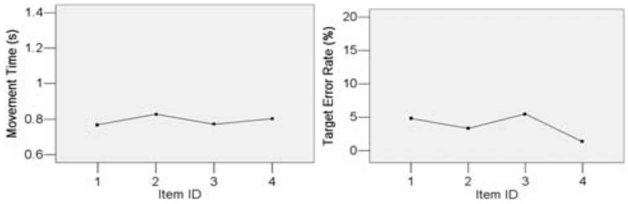

Figure 9. Mean response time and mean error rate for 4 items in the Tilt Menu.

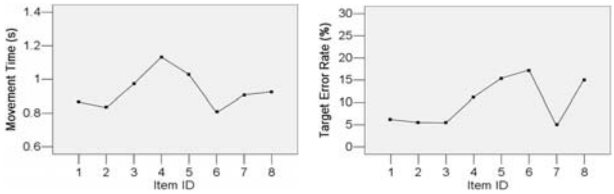

Figure 10. Mean response time and mean error rate for 8 items in the Tilt Menu.

At the same time, we observe that the response time of the item 4 is significantly longer than that of other items. Pair-wise comparisons reveal significant differences between item 4 and other items ($p < .03$)

And error rates of items 4, 5, 6, 8 are significantly higher than that of other items. Pair-wise comparisons reveal significant differences between items 4, 5, 6, 8 and other items ($p < .05$). In addition, participants report that the item 8 is difficult to select because the hand's wrist obscures item 8 for right-handed users.

Similar problems also occur in the Tilt Menu with 12 items, and are even more obvious. Repeated measures analysis of variance show a significant main effect for item orientations in response time ($F_{11,121}$ = 10.220, $p < .0001$) and error rate ($F_{11,121}$ = 171.024, $p < .0001$). Results are shown in Figure 11. We also observe that the response time of the items 1, 2, 9 are significantly shorter than that of other items. Pair-wise comparisons reveal significant differences between items 1, 2, 9 and other items ($p < .05$). And error rates of items 1, 2 are significantly lower than that of other items. Pair-wise comparisons reveal significant differences between items 1, 2 and other items ($p < .05$).

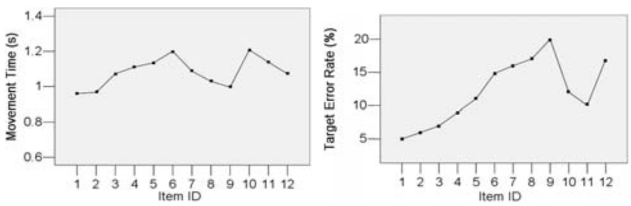

Figure 11. Mean response time and mean error rate for 12 items in the Tilt Menu

Results of experiment 1 show that increasing menu sizes, increases response time and error rate. At the same time, there are significant influences of the item orientation on Tilt Menus with higher breadths, such as menus with 8 or

12 slices. For those menus, selecting "ill-located" items which are below the right side of horizontal axis and upon the left side of horizontal axis (such as items 4, 8 of the Tilt Menu with 8 slices, or items 6, 10 of the Tilt Menu with 12 slices) will result in poor performance.

Experiment 2

The results of experiment 1 provide us with a reasonable understanding of how the parameters of the Tilt Menu would affect its performance in a controlled environment. In experiment 2, we evaluate the benefits gained from Tilt Menu's property of merging command selection and direct manipulation in freeform drawing. Our approach is inspired by and shares some design properties with Guimbretière [12] and Kabbash [16].

We conduct an experiment that includes three types of interaction techniques. Tool palette (TP) uses a sequential command assemblage [12] [16] without merging command selection and direct manipulation. Toolglass (TG) uses an asymmetric dependent command assemblage [12] [16] and merges command selection and direct manipulation by using two hands to perform the task; Tilt menu (TM) uses a sequential command assemblage and merges command selection and direct manipulation using one hand. In experiment 2, we apply the Tilt Menu with 4 items to compare with other techniques. This maintains the consistency with former studies [12] [16] regarding the number of selections available. Moreover, we know that there are no significant influences of the item orientation on the Tilt Menu with 4 items from results of experiment 1.

In experiment 2, marking menu, FlowMenu, control menus and other one handed techniques are not applied as contenders because they cannot merge command selection and direct manipulation in freeform drawing. Like Guimbretière [12] and Kabbash [16], we use the connect-the-dots task for our experiment. The main difference between this experiment and others is that freeform drawing mechanism is used to perform the connect-the-dots task rather than a rubber-band line mechanism [12]. Series of colored dots are presented one by one to the participant in a connect-the-dots task. The participant will select the matching color using one technique and connect with freeform drawing from the last dot in the current path to the new dot once a new dot appears. As for the Tilt Menu, participants are told to perform color selection anytime while drawing, rather than just at the start of drawing. The next dot appears when the connection is completed if the trial is not ended. Typical traces for each technique used in experiment 2 are movement time, error rate and stroke drawn. Different from prior literature [12] [16], the data of the stroke can help us perform further analysis on the precision of task completed in this experiment.

Subjects and Apparatus

Twelve subjects participate in the experiment. To minimize experimental bias due to handedness, we ensure that all participants are right-handed and have normal or corrected-to-normal vision via self-report. No participants were recruited for both experiments. A Wacom 12'×12' touchpad which can simultaneously track a pen and a puck, and a 17' LCD screen with the resolution in 1024×768 pixels are used for the experiment. The gain factor between the tablet and the screen is set to 1.33 based on former studies [12] [5]. In order to avoid collisions between the pen and the puck, the puck tracking is offset by 32mm. This setting is picked based on the setting for best Toolglass performance according to Balakrishnan [5] and Guimbretière [12].

Task and Setting

We use a completely within-subject experimental design. Participants are instructed to complete all trials in this experiment. For all conditions, participants are presented with the same 24 sets of 12 points to connect (11 connections per set). Participants are instructed to draw from the previous dot to the next dot after selecting the correct color as quickly as possible. As for the Tilt Menu, participants are told to perform color selection at any time while drawing, rather than at the start of drawing. At the same time, they are told to follow the dashed line which connects the previous dot to the next dot. Consecutive dots are always of different colors as in [12] and [16]. And participants are told that the connection time is measured from the appearance of a new dot to successful completion of the line, including time to correct any errors in picking the color or connecting the dots.

The screen layout is shown in Figure 12. To improve comparability, participants are given the same dot patterns used by Guimbretière [12] and Kabbash [16]. The 4 possible colors are: red, green, blue, and yellow. The background color is white. All previous dots in the path are rendered in gray and filled. The new target dot is rendered as a circle of the requested color, and the dashed line connected between the last dot and the new target dot is rendered in the requested color as feedback for participants drawing the freeform stroke, to follow the line. Each dot's radius is 11mm. In this set of 24 patterns, the distance between dots varies between 25 and 151mm with the distance distribution. The same 24 sets are used for all conditions. The data set is identical to two former set used by Guimbretière [12] and Kabbash [16].

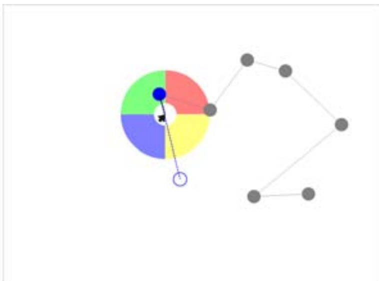

Figure 12. A typical display for experiment 2 with the Tilt Menu condition. The difference from prior literature is that participants connect the dots with freeform drawing rather than rubber-band line.

The detailed settings of three techniques are described as follows. These settings of Tool Palette and Toolglass are the same as the Guimbretière's [12]. The color tool palette consists of 4 buttons, each 16mm by 16mm, with a header 32mm wide and 8mm tall at the top. A typical display of experiment 2 with the Tool Palette condition is shown in Figure 13. The color Toolglass consist of 4 buttons, each 16mm by 16mm, with a header 32mm wide and 8mm tall at the top. The Toolglass is set to 40% transparency for dots underneath to be visible. A typical display of experiment 2 with the Toolglass condition is shown in Figure 14.

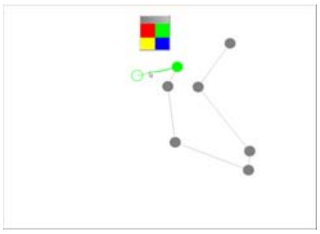

Figure 13. A typical display for experiment 2 with the Tool Palette condition.

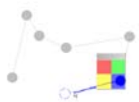

Figure 14. A typical display for experiment 2 with the Toolglass condition.

The radius of the Tilt Menu is 40mm, and the diameter of the center region is 10mm. The Tilt Menu is invoked as soon as drawing initiates (the pen tip pressed on top of the last dot). The menu can automatically follow the pen tip while drawing, until the pen tip is lifted. In any time during this period, participants can use the pen's tilt to perform color selection task. Once the selection is made, the menu immediately disappears. A typical display of experiment 2 with the Tilt Menu condition is shown in Figure 12.

Procedure

Each participant is given the opportunity to practice on 5 sets of 12 dots that are not among the 24 sets used in the experiment. The order of the experimental conditions for each participant is counterbalanced using a Latin square control for order effects. In order to limit carryover effects, the color layouts of the Toolglass, the tool palette and the Tilt Menu are arranged in different orders. After completing all trials, participants complete a questionnaire providing subjective ratings to different aspects of each technique on a scale from 1 (worst) to 7 (best) and providing information about their previous experience with similar systems including whether the technique is fast, enjoyable, error prone, or comfortable to use. Experiments including the questionnaire totally last about 40 minutes for each participant. Participants could only take a rest between sets.

Results

All participants complete all sets. As in Guimbretière [12] and Kabbash [16], the first connection in each set is removed from the data. As a result, we record 240 connections in each of the four conditions for each user.

Unlike former studies [12] [16], we do not analyze the command selection time (the time elapsed between the start of the trial and the time at which the drawing starts) and the drawing time for error-free connections. As for the Tilt Menu, the task time cannot be simply divided into these two phases due to color selection and freeform drawing can be performed simultaneously.

As for the movement time, repeated measures analysis of variance shows a significant main effect for technique type ($F_{2,22}$ = 6.171, p =.005). The descriptive statistics of movement time of three technique types are shown in Figure 15. The mean movement time of the Tool Palette, the Toolglass and the Tilt Menu are 2678.33 ms, 2511.66 ms and 2205.36 ms respectively. We observe that the movement time of the Tilt Menu is significantly shorter than that of the Tool palette ($F_{1,11}$ = 20.385, p = .004), but there is no significant difference found among others. All significance levels for pairwise comparison use a Tukey's test for multiple comparisons.

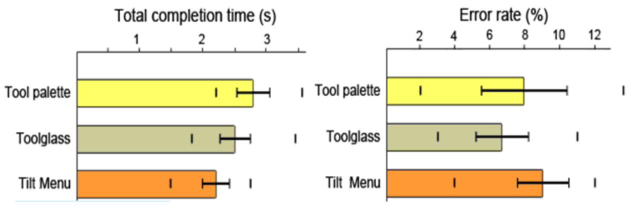

Figure 15. Mean movement time and mean error rate for three techniques. Besides mean values and the standard error, the maximum and minimum values for each series are shown.

As for the error rate, main effect for technique type is not found significant from repeated measures analysis of variance ($F_{2,22}$ = 1.548, p = .228). The mean error rate of the Tool Palette, the Toolglass and the Tilt Menu are 7.95%, 6.65% and 8.97% respectively. All are lower than 10%, and statistically, no significant difference is found among them. All significance levels for pairwise comparison use a Tukey's test.

As for the results of the questionnaire, we collect data using a scale from 1 (Not at all) to 7 (Very much). Besides median values and the 95% confidence intervals, the maximum and minimum values for each series are shown in Figure 16. Results from Wilcoxon signed ranks tests indicate that the Tilt Menu and Toolglass are perceived significantly faster than the Tool Palette (p < .03). No significant difference is found between the Tilt Menu and Toolglass (p = .051). The Tilt Menu is perceived as being more enjoyable than the Tool Palette (p = .023). There is non-significant difference between the Tilt Menu and Toolglass (p = .090). The Tilt Menu and Toolglass are perceived as being more error prone than the Tool Palette (p < .02). There is significant difference between the Tilt Menu and Toolglass (p = .026). Meanwhile, significant differences are not found for the comfortable (p > .13).

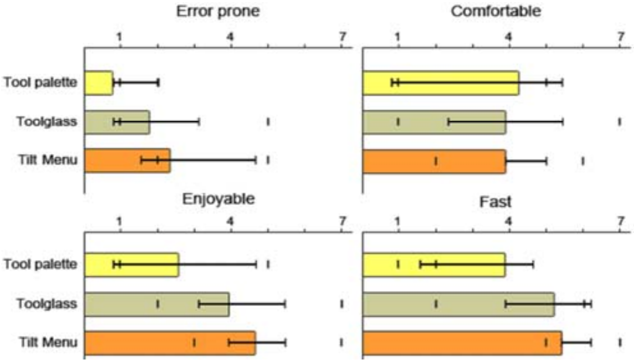

Figure 16. Subjective ratings. Besides median value and the 95% confidence intervals, the maximum and minimum values for each series are shown.

DISCUSSION

Results of experiment 1 show that increasing the menu size increases response time and error rate. This is similar to the influences on the marking menu, because the performance is limited by the ease of articulation of menu selection [18]. As for Tilt Menus with 4 or 8 items, the error rates are less than 10%. However, when using Tilt Menus with 12 items, selection becomes error-prone. Zhao's work [32] suggests that Zone and Polygon menus can be extended in breadth to 16 items, while providing good speed and accuracy. This issue is due to the fact that pen tilt requires more effort than typing or drawing a stroke when performing selection tasks. We know that when using pen tilt to do selection tasks, users have to change the pen holding style dynamically. Franke revealed there are four primary categories of pen-tilt variations [9]. Jagadeesh pointed out that the degree of tilt not only varies with the user but also depends on the stroke the user is writing [15]. Therefore, in some cases, the arbitrary changing of pen-tilt will result in more effort than typing or drawing a stroke.

At the same time, our "item orientation" analysis in experiment 1 indicates that the source of poor performance at higher breadths is partly due to selecting "ill-located" items. Figure 17 shows a demonstration of a right-handed user performing a selection task with a Tilt Menu (menu size = 8). From this, we can infer that item 8 is difficult to select because it is obscured by the right hand. In addition, when right-handed users begin to perform selection task, the azimuth of the pen device is always located in the range of item 8. From the state transition diagram of the Tilt Menu (Figure 4), we can infer that when users begin to perform a selection task, if the pen's length is too long (TCLength is larger than the radius of the center region), this will cause the Tilt Cursor to move out of the center region and hit item 8. Therefore, users have to adjust the pen-holding styles to let the Tilt Cursor back into the center region in order to initiate a selection task. Meanwhile, we should note that for right handed users, the trajectories to item 4, 5 drawn by the pen tail when performing selection task, are normally larger than others items, as shown in Figure 17. This results in longer movement time and higher error rate.

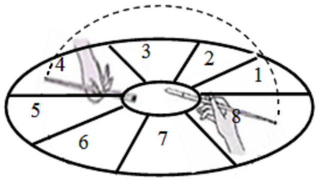

Figure 17. A right-handed user performing a selection task

The influences of "item orientation" described above are similar to the effects of item angle on performance for marking menus reported by Kurtenbach [17]. Thus, when designing a wide Tilt Menu, the most frequently used items should avoid being "ill-located". This would allow some items to be accessed quickly and reliably with tilts, despite the breadth of the menu.

In experiment 2, the diameter of the center region is 10mm. It's due to make it can be comparable to others. We make a consistency of distance between the movement of the pen tail in the Tilt Menu selection task and the movement pen tip in other tasks. In calculating the length of the Tilt Menu, the length of virtual 3D pen is set to 20mm. We can easily get that if the diameter of the center region is set to 10mm, the distance about movement of the pen tail is about 16mm. Meanwhile, it should be pointed out that it is not appropriate to use the Fitts' law to compare time difference between the Tilt Menu with other two treatments. How to accurately measure and predict user behaviors with the Tilt Cursor is an open question. It is our interest to further explore a theoretical model that governs behaviors of the Tilt Cursor.

Results of experiment 2 demonstrate that the improvement of user performance in specific connect-to-dot tasks with the Tilt Menu is significant, compared to the Tool Palette, and is as good as the Toolglass. This is partly due to the center of the Tilt Menu is always being local with respect to the pen tip, and the user can make the pen tilt to select a specific slice without moving the pen tip. Meanwhile, we should note that in freeform drawing, the users' attention is not only on the position of destination item, but also on the drawing path. With the 3D orientation cue of pen provided by the Tilt Cursor [30], participants could adjust their drawing directions dynamically and easily to follow the dashed connection lines.

Meanwhile, results of subjective ratings in experiment 2 reveal that the Tilt Menu is perceived as significantly being error prone than other techniques. Based on the observation of the participant's action in experiment 2, we find that most errors of the Tilt Menu stem from an incorrect color selection. This suggests that when participates use pen tilt to perform selection tasks, they are more easily to be distracted by the drawing task which is simultaneously performed by the same hand.

In contrary, in order to determine whether the freeform drawing task tends to be disturbed by the selection task which is simultaneously performed, we further analyze the precisions of strokes drawn following the dashed connection line with the three techniques. For each technique, we calculate three values ($\overline{Q}$, $\overline{L}$, $\overline{I}$) based on all strokes recorded in successful trails of experiment 2. $\overline{Q}$ is the average deviation of strokes from connection lines, which is calculated by least square fittings. $\overline{L}$ is the average length of strokes. $\overline{I}$ is the average count of strokes' intersection counts with the connection lines. For each technique, we calculate three values ($\overline{Q}$, $\overline{L}$, $\overline{I}$) based on all strokes in successful trails. The results show that:

$$\begin{cases} \overline{Q}_{TiltMenu} < \overline{Q}_{Toolglass} < \overline{Q}_{ToolPalette} \\ \overline{L}_{TiltMenu} < \overline{L}_{Toolglass} < \overline{L}_{ToolPalette} \\ \overline{I}_{ToolPalette} < \overline{I}_{Toolglass} < \overline{I}_{TiltMenu} \end{cases}$$

Note that, for the Tilt Menu, its $\overline{T}$ (the mean movement time), $\overline{Q}$ and $\overline{L}$ are all smaller than those of the others, while its $\overline{I}$ is larger than that of the others. It suggests that, with the Tilt Menu, participants experience a shorter period (shorter time and shorter length of stroke) to adjust the direction to follow the connection line. The time and the length of stroke consumed in adjusting drawing direction can be considered as the response latencies on mapping between visual stimuli (drawing direction, and the connection line) to motor response (adjusting the drawing direction to follow the connection line). With the Tilt Menu, user needs smaller response latencies to create compatible mapping. Based on the results of [30], we know that the Tilt Menu can enhance stimulus-response compatibility of touchpad especially in freeform drawing. Therefore, we can conclude that while performing a freeform drawing task using the Tilt Menu, the drawing task tends to not be disturbed by the selection task which is simultaneously performed.

CONCLUSION AND FUTURE WORK

We present the Tilt Menu, a menu that extends the selection capability of pen-based user interfaces. This new UI metaphor uses the pen's 3D orientation to perform selection tasks without moving the pen tip. This new type of menu has the potential of enhancing traditional one-handed techniques as it supports selection without any pen tip movements, especially in continuous interaction processes, such as freeform drawing, online synchronous sketch recognition, etc. We conduct two experiments to evaluate the performance of the Tilt Menu. Results of experiment 1 show that increasing menu sizes increase response time and error rate. At the same time, there are significant influences of the item orientation on Tilt Menus with higher breadths. Selecting "ill-located" items will result in poor performances. These results could influence the design of applications that use Tilt Menus. Results of experiment 2 show that the Tilt Menu provides a better performance in connect-the-dot tasks using freeform drawing mechanism than the Tool Palette. Results also reveal that while using the Tilt Menu to perform the selection task and the freeform drawing task simultaneously, the selection task is more distractible to the user when compared to the drawing task.

Our studies show that the Tilt Menu is a promising technique for extending the capabilities of pen-based interfaces. However, there are several directions that can be pursued to extend the current work:

Firstly, our experiments reveal that there are "ill-located" items that influence the performance of Tilt Menus significantly for right-handed users. It would be interesting to explore whether the "ill-located" items for left-handed users are symmetric to that of right-handed users. And it would also be useful to improve the design of the Tilt Menu which could avoid the problem of "ill-located" items to the greatest extent.

Secondly, it should be noted that users' performances cannot be successfully modeled with the Fitts' law when performing selection tasks with Tilt Menus. Such tasks could be considered as 3D selection tasks. The Steering Law [2] provides us with a good reference for exploring a new law to model such tasks.

Thirdly, our experiments suggest that Tilt Menus tend to perform better during freeform drawing in comparison to the other two techniques with touchpad. We know that when users sketch on a touchpad, the separation of controller and display leads to mismatches in coordinate systems comparing with other pen-based devices with touch-sensitive screen. Therefore, it would be useful to explore whether the result of this research is still valid for pen devices with touch-sensitive screen.

ACKNOWLEDGEMENT

We thank the financial support from National Key Basic Research and Development Program, No. 2002CB312103, the National High Technology Development Program of China under Grant No. 2007AA01Z158, and National Natural Science Foundation of China, No. 60503054, U0735004 and 60603073. We thank Dr. Mingxuan Chen and Lu Deng for constructing systems for experiments. We thank Ms. Fei Lv and Dr. Yanju Ren for data analysis. We thank Dr. Shengdong Zhao, Dr. Yang Li and the anonymous reviewers for constructive comments.

REFERENCES

1. Adobe Illustrator, http://www.adobe.com/products/illustrator/
2. Accot, J., Zhai, S., Beyond Fitts' Law: Models for Trajectory-Based HCI Tasks. *In Proc. CHI 1997*, ACM Press (1997), 295-302.
3. Arvo, J., Novins, K., Fluid Sketches: Continuous Recognition and Morphing of Simple Hand-Drawn Shapes. *In Proc. UIST 2000*, ACM Press (2000), 73-80.

4. Balakrishnan, R., Baudel, T., Kurtenbach, G., and Fitzmaurice, G., The Rockin'Mouse: integral 3D manipulation on a plane. *In Proc. CHI 1997,* ACM Press (1997), 311-318.

5. Balakrishnan, R., Hinckley, K., The role of kinesthetic reference frames in two-handed input performance. *In Proc. UIST 1999*, ACM Press (1999), 171–178.

6. Bier, E. A., Stone, M. C., Pier, K., Buxton, W., Derose, T. D., Toolglass and magic lenses: The see-through interface. *In Proc. SIGGRAPH 1993*, 73–80.

7. Callahan, J., Hopkins, D., Weiser, M., Shneiderman, B., An empirical comparison of pie vs. linear menus. *In Proc. of CHI 1988*, ACM Press (1988), 95-100.

8. Fitzmaurice, G., Khan, A., Pieké, R., Buxton, W., Kurtenbach, G., Tracking menus. *In Proc UIST 2003*, ACM Press (2003), 71-79.

9. Franke, K., The influence of physical and biomechanical processes on the ink trace methodological foundations for the forensic analysis of signatures, Ph.D. thesis, University of Groningen, The Netherlands, 2005.

10. Grossman T, Hinckley, K. Baudisch, P., Agrawala, M., Balakrishnan, R., Hover widgets: Using the tracking state to extend the capabilities of pen-operated devices. In *Proc. CHI 2006,* ACM Press (2006), 861-860.

11. Guimbretière, F., Winograd, T. FlowMenu: Combining command, text, and parameter entry. *In Proc UIST 2000*, ACM Press (2000). 213–216.

12. Guimbretière, F., Martin, A., Winograd, T., Benefits of merging command selection and direct manipulation. *ACM TOCHI*, *2005*, 12 (3): 460-476.

13. Hinckley, K., Guimbretiere, F., Agrawala, M., Apitz, G., Chen, N., Phrasing techniques for multi-stroke selection gestures. GI, 147-154.

14. Jacob, R.J.K., Deligiannidis, L., Morrison, S., A Software Model and Specification Language for Non-WIMP User Interfaces. *ACM TOCHI*, *1999*, 6(1): 1-46.

15. Jagadeesh Babu V., Prashanth L., Raghunath Sharma R., Prabhakara Rao G.V., Bharath A., HMM-based Online Handwriting Recognition System for Telugu Symbols, *In Proc. ICDAR 2007,* 23-26

16. Kabbash, P., Buxton, W., Sellen, A., Two-handed input in a compound task. *In Proc. CHI 1994,* ACM Press (1994), 417–423.

17. Kurtenbach, G. Buxton, W., The limits of expert performance using hierarchical marking menus. *In Proc. CHI 1992,* ACM Press (1992), 482-487.

18. Kurtenbach, G., Sellen, A., Buxton, W., An empirical evaluation of some articulatory and cognitive aspects of "marking menus". *Human Computer Interaction*, 1993. 8(1): 1--23.

19. Li, J., Zhang, X., Ao X., Dai G., Sketch recognition with continuous feedback based on incremental intention extraction, *In Proc IUI 2005,* ACM Press (2005), 145-150.

20. Li, Y., Hinckley, K., Guan Z., Landay, J.A., Experimental analysis of mode switching techniques in pen based user interfaces. In *Proc. CHI 2005*, ACM Press (2005), 461-470.

21. Odell, D., Davis, R., Smith, A., Wright, P. K., Toolglasses, Marking Menus, and Hotkeys: A Comparison of One and Two-Handed Command Selection Techniques, *In Proc. GI 2004*, 17-24.

22. Partridge, K., Chatterjee, S., Want, R., TiltType: Accelerometer-supported text entry for very small devices, In *Proc. CHI 2001,* ACM Press (2001), 201-204.

23. Pook, S., Lecolinet, E., Vaysseix, G., and Barillot, E., Control menu: Execution and control in a single interactor. In *Proc. CHI 2000,* ACM Press (2000), 263-264.

24. Solidworks, http://www.solidworks.com

25. Ramos, G., Boulos, M., Balakrishnan, R. Pressure Widgets. In *Proc. CHI 2004,* ACM Press (2004), 487-494.

26. Ramos, G., Balakrishnan, R.. Zliding: fluid zooming and sliding for high precision parameter manipulation. In *Proc. UIST 2005,* ACM Press (2005), 143-152.

27. Ramos, G., Balakrishnan, R. Pressure Marks. In *Proc. CHI 2007,* ACM Press (2007), 1375-1384.

28. Saund, E., Lank, E., Stylus input and editing without prior selection of mode. In *Proc. UIST 2003,* ACM Press (2003), 213-216.

29. Tandler, P., Prante, T., Using Incremental Gesture Recognition to Provide Immediate Feedback while Drawing Pen Gestures. *In Proc. UIST 2001*, ACM Press (2001), 18-25.

30. Tian, F., Ao, X., Wang, H., Setlur, V., Dai G., The Tilt Cursor: Enhancing Stimulus-Response Compatibility Based on 3D Orientation Cue of Pen Devices. *In Proc. CHI 2007,* ACM Press (2007), *303-306.*

31. Wigdor, D., Balakrishnan, R. TiltText: Using Tilt. for Text Input to Mobile Phones. *In Proc. UIST 2003*. ACM Press (2003), 81-90.

32. Zhao, S., Agrawala, M., Hinckley, K., Zone and Polygon Menus: Using Relative Position to Increase the Breadth of Multi-Stroke Marking. Menus, *In Proc. CHI 2006,* ACM Press (2006), 1077-1086.