

Combining and Measuring the Benefits of Bimanual Pen and Direct-Touch Interaction on Horizontal Interfaces

Peter Brandl^{1,2}, Clifton Forlines^{1,3}, Daniel Wigdor^{1,3}, Michael Haller², Chia Shen¹

¹Mitsubishi Electric Research Labs
Cambridge, Massachusetts, USA
forlines | shen@merl.com

²Upper Austria University of
Applied Sciences
Hagenberg, Austria
firstname.lastname@fh-hagenberg.at

³University of Toronto
Toronto, Ontario, Canada
dwigdor@dgp.toronto.edu

ABSTRACT

Many research projects have demonstrated the benefits of bimanual interaction for a variety of tasks. When choosing bimanual input, system designers must select the input device that each hand will control. In this paper, we argue for the use of *pen and touch* two-handed input, and describe an experiment in which users were faster and committed fewer errors using pen and touch input in comparison to using either *touch and touch* or *pen and pen* input while performing a representative bimanual task. We present design principles and an application in which we applied our design rationale toward the creation of a learnable set of bimanual, pen and touch input commands.

Author Keywords

Bimanual input, pen and touch, self revealing gestures.

ACM Classification Keywords

H.5.2 INFORMATION INTERFACES AND PRESENTATION
(e.g., HCI): User Interfaces - Input devices and strategies (e.g., mouse, touchscreen)

1. INTRODUCTION

The benefits of bimanual interaction have been investigated in numerous research projects [4, 7, 8, 9, 14, 32]. By leveraging input from both hands, system designers can increase the input bandwidth from their users and add rich and natural interactions to their applications. When designing for bimanual input, system designers must choose among the many input devices available for each hand. Comparisons among various input devices (such as mice, pucks, stylus, and touch-tables) are plentiful [16, 18, 21, 30]. Taken as a whole, this body of research indicates that individual input devices excel in certain measures and lack in others.

For example, multi-touch interactive surfaces [13, 28, 32] have the strong advantage that no intermediary input device is required. For this reason, this type of direct, “under-the-finger” input device is often called “natural” and “intuitive” when compared to a mouse or stylus. Additionally, by sensing multiple points of contact, these devices allow for complex input [32, 36]. On the other hand, occlusion and finger size hamper accurate touch input in a graphical interface. In contrast, a computer stylus provides a higher-level of input accuracy, but typically only a single point of input. While choosing any *one* particular input device requires weighing these types of tradeoffs, bimanual input allows us to

design input commands using different input devices with the dominant and non-dominant hands. Previous work in this area, along with our observations and experimentation, has convinced us that by combining dominant-hand pen input with non-dominant hand touch input, we can effectively harness the benefits of both pen and touch input while avoiding many of their pitfalls.

In this paper, we first survey the related work in the field of bimanual interaction, and then describe a set of design principles based on this work. We further describe our graphical editing application (cf. Figure 1). This application was developed based on the principles described in past work, as well as leveraging the strengths of both pen and touch input. Finally, we present the results of a laboratory experiment, in which the combination of pen and touch input outperforms bimanual pen and bimanual touch input for a representative task.



Figure 1. A designer sketching using our prototype.

2. RELATED WORK

Bimanual interaction has been investigated in numerous research projects and formal studies using a variety of input devices [1, 3, 20, 21, 24, 31]. The previous work on bimanual input can generally be divided into two categories: the first category defines or extends models and frameworks for bimanual input, and the second one applies those models and frameworks.

2.1 Models and Frameworks

Most work in bimanual interaction has been influenced by Guiard's Kinematic Chain model [17], which proposes general principles for asymmetric bimanual actions. During two-handed interaction, both hands have different roles that depend on each other with respect to three rules: the dominant hand (*DH*) moves within the frame of reference defined by the non-dominant hand (*NDH*); the sequence of motion generally sees the *NDH* setting the reference frame prior to actions with the *DH* being made within that context; and that the *DH* works at a higher level of precision than the *NDH* in both spatial and temporal terms. Follow-up research has extensively investigated different facets of these hypotheses, such as the importance of visual and kinesthetic feedback for bimanual tasks [1, 3] and differences between symmetric [24, 31] and asymmetric [10, 11, 18] bimanual input.

Kabbash et al. [21] studied four techniques for performing a compound drawing and color selection task using a unimanual technique, a bimanual technique in which each hand controlled independent tasks, and two bimanual tasks where the DH depended on the NDH. They suggest that asymmetric, dependent tasks are most effectively performed using two hands. Several research projects have sought to apply these findings and to investigate interaction design and input devices for bimanual tasks.

2.2 Interaction Design and Input Devices

Different input devices have been evaluated regarding their suitability for bimanual tasks. The use of bare hands for gestures and self-revealing tasks has been studied by Kruger [22] in VIDEOPLACE. Matsushita et al.'s. Holowall includes bimanual object manipulation [28]. SmartSkin [32] is an interactive surface enabling bimanual interaction for different tasks, such as map panning and zooming. Several projects have explored the benefits of two-handed control [7,14,16,36] using DiamondTouch [13].

Kabbash et al. [21] performed a comparison among a mouse, a trackball, and a stylus for bimanual tasks. Their findings support Guiard's claim that the NDH is best suited for imprecise tasks. Forlines et al. [16] conducted a study that compared bimanual mouse and touch input on interactive tabletops. A combination of a PDA in the NDH and a mouse in the DH was investigated by Myers et al. [30].

Matsushita [27] and Yee et al. [1] implemented mobile devices supporting pen and touch input. In both cases, touch input complemented pen input. Cutler et al. investigated the use of a glove and pen on the Responsive Workbench [12]. They found that the combination of a glove for the NDH and a stylus in the DH worked best for asymmetric tasks that reflected the natural qualities of each input device. The stylus with its thinner tip and more precise point of touch fits better for high-precision tasks.

Researchers have explored bimanual interaction for a variety of tasks that can be performed more efficiently compared to a sequential single-handed input. Potential tasks include menu control [4, 6, 23, 35], desktop interaction such as selecting [8], scrolling [9, 26] and cursor control [5, 14], map navigation [18, 32, 33, 34] and sketching [7, 15, 23, 1].

Hinckley et al. [18] explored the performance of puck and stylus as well as touchpad and *TouchMouse* combinations for bimanual interaction. They found bimanual benefits for map navigation tasks. Formerly sequential actions were chunked by the simultaneous use of two devices were therefore performed more quickly. Kurtenbach et al. [23] tested two-handed interaction with Toolglass menus with a graphics editing program. They used WACOM tablets with two pucks as input devices to evaluate their design approach that aims at maximizing the screen space for application data while providing an increasing quality of input.

The contribution of our work is to leverage insights of previous bimanual-input related research in developing a framework for system designers. This framework is intended to guide the development of interaction using combinations of pen and direct touch input for tabletop interactions. Compared to relatively small surfaces such as Tablet PCs or graphic tablets, the necessity of visually linking the tasks of both hands on a tabletop becomes increasingly important. Switching the attention between left and right hands results in highly sequential performance and neutralizes or reverses the advantage of bimanual interaction [3, 21], a design based on both descriptive principles and predictive models is especially demanded for large surfaces.

3. DESIGN PRINCIPLES

We sought to establish a set of design principles intended to guide developers of bimanual user interfaces. This set is based on an exploration of the possible combinations of bimanual input when each hand may be used for either pen or touch input, aided by previous work in bimanual systems. By investigating the possible pairs of input actions, one performed by the DH and one by the NDH, as well as the type of and order in which these commands are performed, we have developed a set of principles that we believe combine the best qualities of pen input and touch input into a single system.

3.1 Input Choices for Both Hands

The general structure we propose for the categorization of input variations is shown in Figure 2. Both, the DH and the NDH have the same set of input possibilities – pen input, touch input, or no input. When using a pen for input (Figure 2, *left branch of tree*), researchers and system designers typically distinguish between *inking* and *command stroking* modes: referring to the usage of a pen for writing or drawing, and for making commands in the second. A barrel-button can be used to delimit these two modes, as can gestural delimiters [18]. Command strokes are interpreted as either point-based interaction (i.e. mouse-like, point-and-click commands) or gestural strokes (e.g. handwriting input).

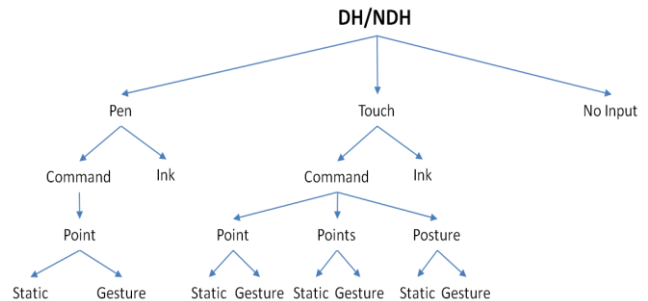


Figure 2. Input categorization. Both the DH and NDH can perform one type of input in a bimanual action.

When using touch input (Figure 2, *center branch*), single-finger commands are often interpreted as point-based interaction (i.e. mouse-like interaction). A benefit of touch input with multi-touch devices is the ability to sense and handle multiple points of input, or even different hand postures. Postures can be recognized as commands themselves or moved over time creating high-bandwidth gestures.

For designing bimanual commands, two different inputs are combined to infer a task or operation. Input possibilities are stated in the tree as leaves. One input comes from the dominant and one from the non-dominant hand.

3.2 Pros and Cons of Pen and Touch Input

We considered the pros and cons of each technique to motivate the assignment of different input combinations for different tasks (cf. Table 1). We wished to combine the positive qualities of both input mechanisms, while avoiding their pitfalls. According to previous studies on the role of the dominant and non-dominant hands [12, 21], we propose using a pen for precise input with the DH and direct touch for intuitive, high-bandwidth touch commands with the NDH. For example, inking is a typical pen task, as touch input suffers from larger occlusions and lower touch precision. Cutler showed that this use of pen and touch performed better than two-handed touch [12].

They found that, especially for asymmetric tasks, the benefit of a distinguished pen point for fine grain gestures and the intuitive use of coarse hand gestures exactly mirrored the asymmetric distribution of labor described by Guiard.

	PROS	CONS
PEN INPUT	Less accidental input than touch	Only one input point
	Precise touch point / High sensing resolution	Separate device
	Familiar tool that leverages users' experience	Less occlusion emphasizes parallax
	Less occlusion of targets than touch	
TOUCH INPUT	Multiple points of input, high # of degrees of freedom (high bandwidth)	Occlusion by hands and fingers
	Use with low attention	Low touch point preciseness (fat fingers)
	"Natural"	
	No extra input device to manage	

Table 1. Advantages and disadvantages of pen and touch as input devices for tabletops.

3.3 Sequencing of Commands

We have already discussed how each hand can issue one of several input commands and that the combination of direct touch and pen input offers the tantalizing opportunity to take advantage of the strengths of each device. Additionally, we explored the sequencing of the DH and NDH actions. A bimanual task can be started by either hand, and the sequence of the start of the paired input streams can set the context for the further action. Wu et al. [36] refer to this concept as the gesture registration phase that defines the beginning of every gesture operation and therefore sets the context for subsequent interactions. They describe a system in which a stylus can be either treated as a writing device or a pointer depending on the mode set in the gesture registration phase.

When the sequence of bimanual actions is relevant, a pen's input preceding a touch gesture is different to a touch gesture performed prior to a pen point. According to their temporal occurrence, we distinguish three different types of sequences: sequential, overlapping and simultaneous (cf. Figure 3).

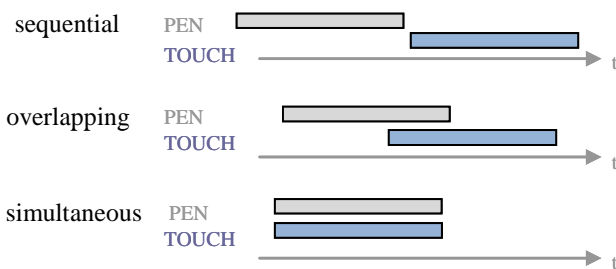


Figure 3. Three causal sequencing of commands.

3.4 Coupling and Decoupling of Interactions

In terms of sequencing, we dynamically and systematically couple and decouple the input of the two hands. For example, a task that can be performed with the NDH could be supported by the DH to extend the functionality or increase the accuracy. The NDH therefore sets the modal reference frame in which the DH will be acting. We add the input of the DH if necessary (couple) and proceed with single NDH input if this is sufficient (decouple). Coarse positioning of an object, for example, can be achieved with the NDH; for a final accurate placement, the DH can be coupled to add high precision information.

3.5 Self Revealing Gestures

Effective feedback is critical in a system that accepts bimanual input consisting of points, postures and gestures. We propose a new type of tooltip that is based on the two level concept of invoking mechanism and consequence. There are different ways of invoking an action: through pen input or with direct touch that allows additional gestures. The user sees these possibilities in the first row of the feedback panel. In the second row, he sees the consequence of each action. Thus he always knows what he can do as a next step and what the consequence will be. We use additional tooltips that are placed near an interactive item in the scene, a button for example. The same tooltip is also shown in the feedback panel. The tooltip in the scene gives additional information about the position where the action shown in the feedback panel has to be invoked.

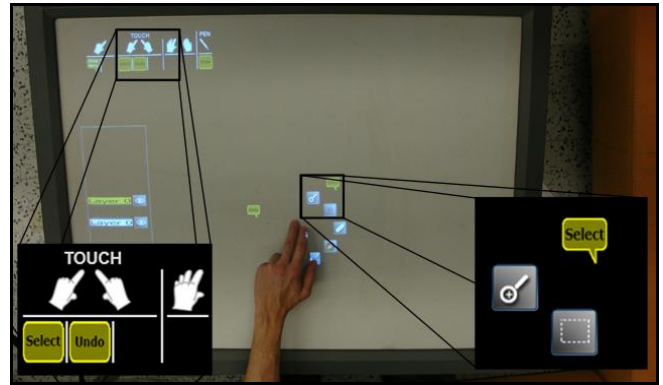


Figure 4. Always visible user feedback shows possible options in each state (left). Tooltips in the scene show positions for command invocation (right).

Our implementation of the mechanism and consequence tooltip concept with position relevant information is depicted in Figure 4. The always visible information panel on top of the screen (cf. Figure 4, left) reveals possible actions as a combination of invoking mechanism and consequence. If applicable, balloon shaped tooltips in the scene show the positions for specific commands (cf. Figure 4, right). Continuous actions that are currently performed are shown as balloon tooltips with inverted text color in the feedback panel. Matching balloons in the scene will then also show an inverted text.

4. PROTOTYPE APPLICATION

We built a proof-of-concept prototype application, based on our design principles, that wraps Adobe Photoshop, a popular image editing application. Commands are issued through combinations of pen-and-touch input on a digital tabletop. We note that the goal of this prototype is to prove our assumptions in a real world scenario, whereas the concepts address a broader field and could be as well applied to different areas.

4.1 Enabling Technology

To enable direct touch and pen interaction we used a DiamondTouch [13] table to sense touch interaction and Anoto [1] technology for pen input. To achieve a co-incident touch and pen sensing surface, we augmented a DiamondTouch table with a transparent sheet on which we printed an Anoto dot pattern (cf. Figure 5). For our prototype setup, we used multiple Bluetooth streaming pens from Maxell [29]. With this combination of input devices, both the touch-table and the pens provided unique IDs.

The prototype is implemented in C# using the Windows Presentation Foundation (WPF) graphics engine. Our application renders a transparent layer on top of Adobe Photoshop CS3 which we basically use for handling images and sketching tasks. The communication between our application and Photoshop is accomplished through a combination of .NET Automation functions, mouse emulation and generated keyboard shortcuts that invoke the appropriate commands in Photoshop. We hide all Photoshop graphical user interfaces and show how a subset of them can be substituted with our bimanual interaction techniques.

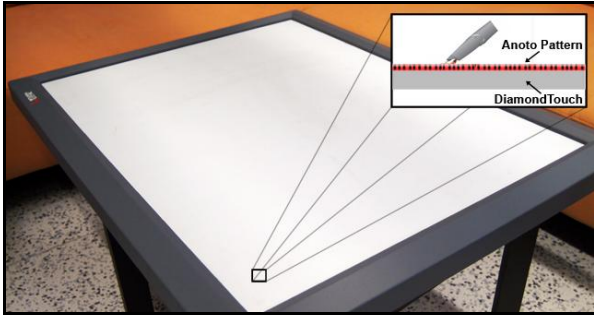


Figure 5. Direct touch and pen tracking surface as a combination of Anoto and DiamondTouch technology.

4.2 Application

Our prototype application shows selected tools that are taking advantage of the strengths of the pen and touch combination and their different roles in bimanual tasks. The functionality covers basic drawing commands in graphics application, like sketching, color picking, brush sizes and eraser tools. Beyond that, further interaction includes scene management like zoom and pan, different kinds of selections and a history tool.

4.2.1 Sketching

In sketching, the precise pen is used as a DH drawing tool while the NDH is free to manage the drawing area through natural panning and zooming touch interaction (cf. Figure 6, *left*). We implemented the zooming feature according to the two point stretch and squeeze technique seen in [19]: touching the surface with two fingers enters the zooming state; the distance between the two fingers defines the level of magnification. More than two fingers touching the surface define the pan state, the drawing area moves with the fingers. As the gestures for entering these states are simple, we support fast scene management without the need to shift the focus from the pen's point of sketching to the control hand.

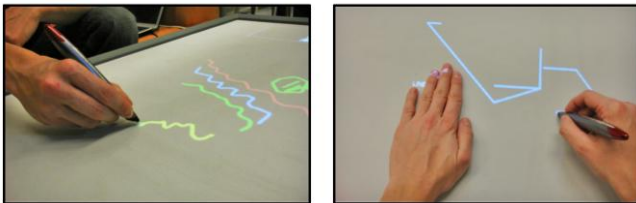


Figure 6. Free-form (left) and straight-line sketching (right).

In addition, we implemented a novel variation of a spring loaded mode for the pen (cf. Figure 6, *right*). If the user touches the tabletop with his flat NDH, the pen remains in drawing mode but is constrained to draw straight lines. The flat hand gesture is easily performed, it can be used anywhere on the surface and directly affects the drawing of the pen. In this sense, it fits well to the coarseness of touch input that sets the mode for the precise drawing tool.

4.2.2 Menu

Using only one finger of the NDH, a menu at the finger's position is shown (Figure 7). Moving the finger drags the menu, once the finger is lifted, the menu fades out. This is an example of a transition from static touch point to point gesture input. Options in the menu can be selected by simply clicking the buttons or stroking over them with another finger of the NDH; this shows a static point to static posture transition. The interaction with the menu can be performed with one hand, the positioning of the menu and selections in the menu are designed for coarse touch input. Moreover, the placement of the menu items is easily learned which aims at the kinesthetic memory and fast repeatability. The layout of the menu is designed to position the buttons left or right of the NDH finger, depending on the handedness of the user.

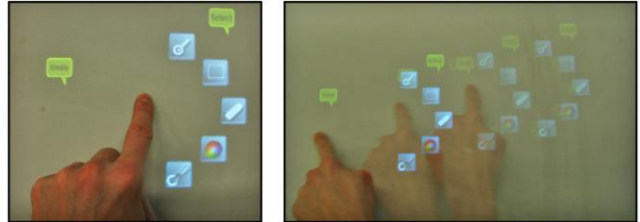


Figure 7. The menu command. One finger touch shows the menu (left). Dragging the menu (right).

4.2.3 History Tool

The history tool can be used to undo one or more steps depending on the dwelling of the hand's posture. Tapping with the NDH to the opposite side of the menu causes a single undo. The history tool appears for a second and shows one-step-back. This mode can be selected very fast and without even looking at the menu as the whole side serves as responsive area. Holding this hand posture enters the multiple-steps-undo mode and keeps the history tool visible. The history tool can be positioned with the NDH while the pen sets the number of undo steps. This interaction fits well to the high precision DH input of the pen and the gesture action of the NDH touch (cf. Figure 8).

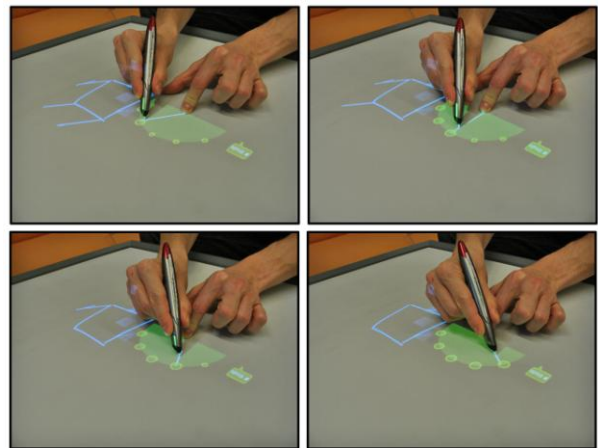


Figure 8. A two finger touch shows the history tool, the precise pen defines the number of undo steps.

4.2.4 Color Picker

The current drawing color can be changed through a HSV color picker (Figure 9). Our implementation handles two simultaneous inputs that control the H and the SV component of the color: the pen offers precise picking of the H-value, one finger selects the SV-value and two fingers drag the color picker.

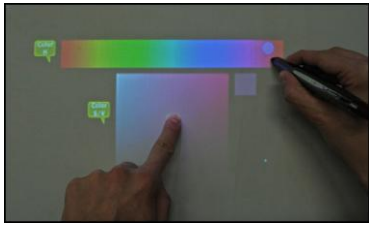


Figure 9. Bimanual color picking. Fine selection with the pen and coarse two-dimensional touch action.

4.2.5 Selection

Our prototype provides three different kinds of selections (rectangular, polygonal and lasso) that illustrate concepts of bimanual sequential dependent interaction. Rectangular selection areas can be defined in two different ways, two-point simultaneous placement or corner placement combined with adjustable dimensions [9]. This choice is made through the sequencing of pen and touch actions. For a two point rectangular selection, the user first defines one corner with his finger. Then he selects the two point option from the menu with the pen. By clicking this button, he sets the diagonal second corner of the rectangle at the pen's position (cf. Figure 10). With this command sequence, one corner for the rectangle can be located first, the option for rectangle selection is chosen from the menu that appears on that location afterwards, and finally the selection rectangle can be set by adjusting the second corner without ever losing the position of the first corner. In this case, we are not aiming at the different accuracies of pen and touch but rather at the simultaneous use for a bimanual task. Another way of defining a rectangle selection is to choose the option with the pen from the menu without a simultaneous single finger touch on the surface. In this mode, the pen is used to set one corner of the rectangle and subsequently stretch the second corner to define the dimension. Again, the pen is used to perform precise actions.

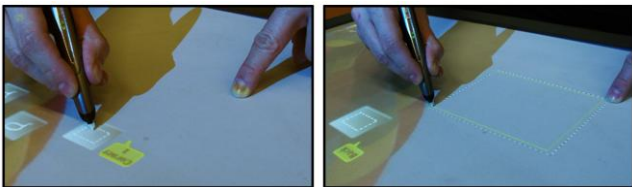


Figure 10. Bimanual rectangular selection with pen and touch. The option is selected from the menu with the pen while the finger already defines one corner position (left). Stretching the rectangle selection with the pen controlling one corner and touch setting the other (right).

For polygonal selections, the points of the polygon shape are defined by the pen while the NDH confirms their position with a single finger touch (cf. Figure 11). With this technique, the points can be placed very accurately with the pen whereas the hand's touch can be performed at any arbitrary position. In contrast, pens with integrated buttons may suffer from a small jitter during a button press. Releasing the pen causes the polygon shape to be closed immediately. The lasso selection is performed solely with the pen.

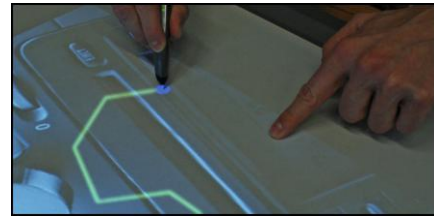


Figure 11. Polygon selection featuring precise waypoint definition with the pen and intuitive touch confirmation.

4.2.6 Cut/Copy Paste

After a selection is finished, cut and copy actions can be performed on the selected region. Again, we consider the NDH for this task as we can use gestures to provide fast access to all possible modes. Tapping with one finger on the selected region results in a cut action, whereas two finger interaction means copy. Immediately after performing the cut or copy action, the new area can be further positioned with the NDH (cf. Figure 12). Moving the object with the hand is a very intuitive action that can be carried out fast with average preciseness. But when it is required to achieve pixel accurate results, touch is not sufficient in terms of resolution, occlusion and jitter.

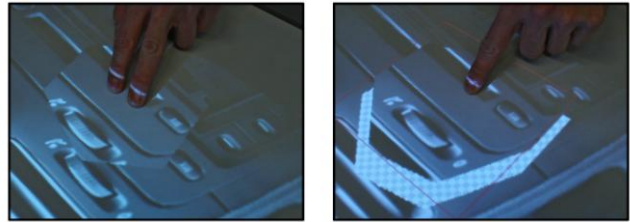


Figure 12. Copy (left) and cut (right) action.

For this reason, we introduce a novel technique that benefits from the advantages of the pen and touch input devices. We already introduced the underlying concept of dynamically and systematically coupling and decoupling of the two hands. Therefore, we propose the use of the pen in the DH to gain additional preciseness that can be controlled from any position on the table. Once the pen is used simultaneously with the touch to position an object, visual connection between the two hands is shown in form of four lines connecting the pen's point with the corners of the object's bounding box (cf. Figure 13).

The pen's movements are directly applied to the selected object to control subtle transformations; the touch is locked meanwhile to prevent jitter influence. Once the pen is lifted (decoupled), the touch is again controlling the object. We note that according to our design considerations, the pen can be used if necessary to add preciseness, but the action of positioning itself still can be carried out with the NDH touch alone.



Figure 13. Coarse positioning of a selection with the hand (left and center). Pen in second hand allows pixel accurate transformation (right).

5. LABORATORY EXPERIMENT

Previous work has argued for the advantages and disadvantages of pen and touch combinations; however, they have not been investigated in a laboratory experiment. To address this issue, we conducted an experiment that explores the possible assignments of input devices to each of the hands and their effects on efficiency, fluidity, and user preference. Our goal was to understand the differences among the possible input device pairings for a representative task. The experimental task was carefully chosen to tease out the differences between the input device-to-hand pairings, while maintaining ecological validity.

5.1 Participants, Apparatus and Task

Twelve subjects were recruited for our study through an on-line community bulletin board, and paid \$20 (USD) for participating. Seven were male and five were female, and their ages ranged from 20 to 50 years old. Eleven of the 12 subjects were right-handed.

Our experimental task consisted of solving and navigating through mazes by drawing a path from a green start marker to a red finish marker (Figure 14). These mazes were designed so that participants had to magnify the maze in order to successfully follow its paths without colliding with the maze walls as well as zoom out in order to plan a path through the maze that would reach the goal. We believe that the maze solving experimental task has a high-level of ecological validity because it matches many graphical editing operations in which a user repeatedly switches back and forth between detailed editing at a high-zoom level and contextual verification of the changes at a low-zoom level (such as when masking a region of a high-resolution image for clipping). In essence, this is a traditional *path following / tunneling* task with the added element of route planning.

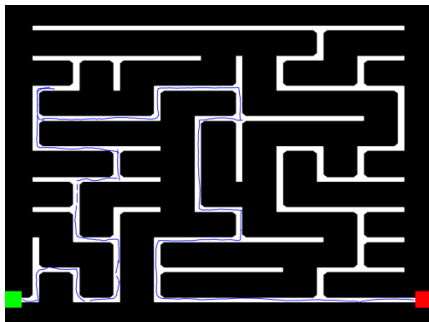


Figure 14. A maze from our experiment with the participant's path stroked through the tunnels.

An error was recorded whenever the participant's stroke intersected with the black walls of the maze. When this collision occurred, a buzzing sound was played, the subject's stroke changed color from blue to red and was stopped. To continue, the participant had to pick a 10 by 10 pixel *continue target* that was displayed at the last valid position before hitting the wall. Upon returning to the white path of the maze, the stroke returned to blue. To complete the maze, the participants had to draw one continuous stroke; each time they lifted the pen or drawing finger, the continue target at the end of the stroke had to be picked to proceed. Each participant controlled the testing application using three different input techniques. Each of the three techniques was a bimanual input technique in which the dominant hand created strokes through the maze and the non-dominant hand zoomed and panned the maze itself. The techniques differed in terms of what input device the dominant and non-dominant hands controlled.

In the first technique, a participant held two pens, one in each hand. While their dominant hand's pen created strokes through the

maze, their non-dominant hand controlled a simple marking menu from which they could zoom in/out and pan the maze. The zoom option was selected with a stroke over the right 90 degree region in front of the pen, the pan selection was performed in the left 90 degree region. In the zooming case, a forward motion would zoom into the scene, whereas a backwards motion zoomed out. After learning the left/right assignment for pan/zoom, this marking menu could be used without paying visual attention. We refer to this technique as *Pen/Pen*.

In the second technique, the participants held a pen in their dominant hand, which they used to create strokes in the maze, while they performed two simple gestures for zooming and panning with their non-dominant hand. Two fingers spreading apart or pulling together would zoom in or out respectively, one finger panned the maze. We refer to this technique as *Pen/Touch*. The C/D gain for zooming was the same for the pen's marking menu option and the direct touch stretching gesture. The mapping coefficient was multiplied by a fixed value to achieve a larger zooming effect with less motion.

Our third and final input technique, *Touch/Touch*, combined the non-dominant hand gestures for pan and zoom from the *Pen/Touch* condition with index-finger stroking performed with the participant's dominant hand.

5.2 Hypotheses

Our hypotheses, the confirmation of which will validate our hand / input device pairings, were as follows:

H1: Participants will complete the mazes in less time while using the Pen/Touch technique than when use the Pen/Pen or Touch/Touch techniques.

H2: Participants will commit fewer errors while using the Pen/Touch technique than when use the Pen/Pen or Touch/Touch techniques.

H3: Participants will prefer the Pen/Touch condition over the other conditions.

5.3 Design

We used a within-participant, repeated measures design for our study, with each subject completing 10 mazes using each of the 3 input techniques. The order of the three techniques was balanced between participants. All participants completed the same 30 unique mazes, and maze / technique pairings were balanced. Participants were given instructions before using each technique, and were asked to practice the technique on two practice mazes before starting the experimental trials. In short, our design was:

$$12 \text{ participants} \times 3 \text{ Input Techniques} \times 10 \text{ mazes} = 360 \text{ trials}$$

5.4 Results

5.4.1 Time Analysis

The time of a trial was recorded as the time between the participant's click of the start button that was shown before each maze and their successful crossing through the "finish" rectangle at the end of the maze. A repeated-measures ANOVA shows that there was a significant difference among the three input techniques ($F_{1,11} = 10.70$, $p < 0.01$), thus confirming hypothesis H1. On average, our participants successfully completed each maze in 42.6s, 36.5s, and 52.7s for the Pen/Pen, Pen/Touch, and Touch/Touch conditions respectively.

5.4.2 Error Rate Analysis

In our study, an error was recorded whenever the participant's stroke collided with one of the walls of the maze.

When this occurred, an error sound was played, the color of the participant's stroke changed from blue to red and could not be continued until the small recover rectangle at the last valid stroke position was picked. Upon reentering the white path of the maze, the sound would stop and the stroke color would return to blue.

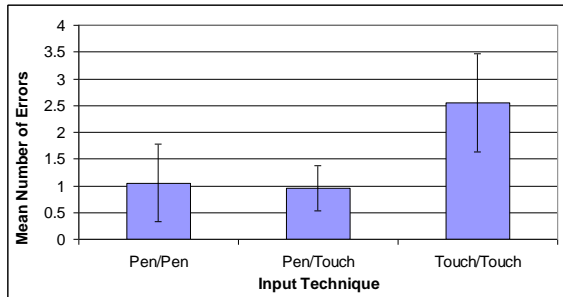


Figure 15. The mean number of errors committed during each maze for each of the three input techniques. Error bars represent 95% confidence interval.

A repeated-measures ANOVA suggests that there is a significant difference among the average number of errors committed by our participants while using each of the three input techniques ($F_{1,11} = 11.6, p < 0.01$). On average, participants committed 1.05, 0.95, and 2.55 errors per maze for the Pen/Pen, Pen/Touch, and Touch/Touch conditions respectively. A post-hoc comparison of means shows a significant difference between the Touch/Touch and both of the other two input conditions in respect to error rate. Figure 15 shows the average number of errors per maze for each input technique.

5.4.3 Preferential Results

At the end of each session, we asked our participants to rank the three techniques in terms of ease of use, accuracy, and overall preference. Table 2 shows the mean rank and standard deviations for each of the three techniques for each of the three measurements (lower numbers indicate a higher level of preference). These results support hypothesis H3, in that our participants seemed to indicate a strong preference for Pen/Touch input over the other two techniques, with 10 of our 12 participants ranking *Pen/Touch* as highest in terms of overall preference.

Table 2. Mean (StDev) rankings for each input techniques.

	<i>Touch/Touch</i>	<i>Pen/Pen</i>	<i>Pen/Touch</i>
Overall Preference	2.50 (0.67)	2.33 (0.65)	1.17 (0.39)
Ease of Use	2.50 (0.67)	2.25 (0.75)	1.25 (0.45)
Accuracy	2.83 (0.39)	1.92 (0.67)	1.25 (0.45)

5.5 Discussion

In need of investigation is an accounting of the observed differences in task time between our three input techniques. While the number of errors committed certainly accounts for some of the difference in trial times, they do not fully explain it. In addition to recording the trial time and number of errors committed during each trial, our testing application also recorded the number of *zoom* and *pan* operations as well as the number of times that a participant lifted the pen (in *Pen* conditions) or index finger of their drawing hand (*Touch/Touch*) from the table. These numbers provide details allowing us to provide additional insights from the observed differences in completion time.

An examination of the number of zoom operations provides further insights. An ANOVA shows that each of the input techniques had a significantly different number of zooms

($F_{2,22} = 23.0, p < 0.001$). On average, participants zoomed 1.62, 1.89, and 7.91 times per maze for Pen/Pen, Pen/Touch, and Touch/Touch respectively (Figure 16). The much larger number of zooms in the Touch/Touch condition is explained by the lack of precision of the finger for drawing input: participants zoomed in to draw, then back out to gain context in navigation. At the other extreme, participants zoomed significantly less often in the Pen/Pen than in the Pen/Touch condition, despite the identical drawing device. We attribute this difference to the increased awkwardness of using the pen-based menu versus gestures.

Additional timing information can be deduced by examining the number of panning operations during each trial (Figure 16). The mean number of pans was significantly different in each of the input conditions: 0.93, 2.14, and 11.37 for Pen/Pen, Pen/Touch, and Touch/Touch ($F_{2,22} = 23.8, p < 0.001$). Panning is positively correlated with zooming, since zoomed-in mazes require more pans to traverse the space, while requiring frequent zooms in and out to gain context and to draw strokes. As with the mean number of zooms, we see a reduction in the number of pans in the Pen/Pen condition as compared with the Pen/Touch condition. The reason for this result lies in the behavior of the participants, who tried to avoid using the marking menu in the Pen/Pen condition, while hesitating less to perform gestures for zooming in the Pen/Touch condition. Although the marking menu offered only two options to select from and the selection gesture could be learnt after the first usage, we observed a constant focus shift when the participants used the menu. This behavior was not found in the case when they used touch gestures to zoom and pan.

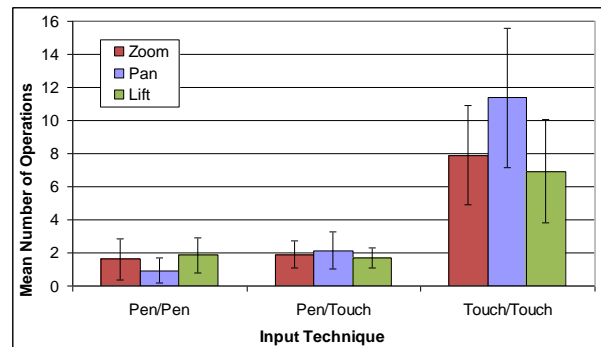


Figure 16. Mean pan, lift, and zoom actions per trial.

The final measurement that helps to explain the observed differences in task time among the input techniques is the number of times that a participant lifted their dominant hand from the tabletop. Again, we see a significant difference among the input techniques ($F_{2,22} = 15.38, p < 0.001$), with a large difference between the Touch/Touch input technique (6.93 lifts/maze), and both the Pen/Pen (1.86 lifts/maze) and Pen/Touch (1.68 lifts/maze) techniques (Figure 16). The fact that a significant higher number of lifts occurred with the Touch/Touch technique seems to be caused by two reasons. First, during the zoom and pan operation, most participants lifted the pen or drawing finger. They could have left the finger or pen on the last drawing position while zooming without causing an error. During panning, this would have resulted in a similar effect of dragging a sheet of paper under a pen. Nevertheless, they felt more comfortable to lift the finger or pen during these actions. Second, due to larger occlusion areas in the Touch/Touch scenario, the stroke was frequently interrupted for hand positioning reasons. Taken together, the clear evidence in support of our hypotheses and these additional details provide strong validation for our assignment of input devices to the hands.

6. CONCLUSION

In this paper, we presented a survey of prior work on bimanual input which led us to a set of principles for the design of two-handed input techniques. These principles included the assignment of pen and touch when considering bimanual input on a horizontal display. To justify this guideline, we conducted an experiment in which three different input combinations for two-handed interaction on horizontal surfaces were tested: *touch and touch*, *pen and pen* and *pen and touch*. The results of this experiment suggest that *pen and touch* input is superior in terms of speed, accuracy, and user preference. As a further validation of our design principles, we implemented a prototype graphical editing application, which includes a new method of teaching bimanual gestures. Considering early feedback we gathered about our prototype implementation, we are confident in the application of our design principles for the creation of future two-handed interactions.

7. FUTURE WORK

We would like to further investigate the learnability of our system in repeated sessions with the same users. We are excited about the opportunities of gathering valuable insights from a user customizable system; therefore a next step would be to enforce the development towards a more flexible application. Accordingly, we would like to explore the mechanism and benefits of an adjustable feedback concept, extending our proposed solution to better accommodate users' requirements.

8. REFERENCES

1. Anoto. <http://www.anoto.com>
2. Balakrishnan, R. and Hinckley, K. (1999). The role of kinesthetic reference frames in two-handed input performance. *UIST 1999*. p. 171-178.
3. Balakrishnan, R. and Hinckley, K. (2000). Symmetric bimanual interaction. *CHI 2000*. p. 33-40.
4. Balakrishnan, R., Patel, P. (1998). The PadMouse: facilitating selection and spatial positioning for the non-dominant hand. *CHI 1998*. p. 9-16.
5. Benko, H., Wilson, A., and Baudisch, P. (2006). Precise selection techniques for multi-touch screens. *CHI 2006*. p. 1263-1272.
6. Bier, E. A., Stone, M. C., Pier, K., Buxton, W., and DeRose, T. D. (1993). Toolglass and magic lenses: the see-through interface. *SIGGRAPH '93*. p. 73-80.
7. Butler, C.G., Amant, R.S. (2004). HabilisDraw DT: A Bimanual Tool-Based Direct Manipulation Drawing Environment. *CHI 2004*. p. 1301-1304.
8. Buxton, W., and Myers, B. (1986). A study in two-handed input. *CHI 1986*. p. 321-326.
9. Casalta, D., Guiard, Y., Beaudouin-Lafon, M. (1999). Evaluating two-handed input techniques: Rectangle editing and navigation. *CHI 1999*. p. 236-237.
10. Chatty, S. (1994). Extending a graphical toolkit for two-handed interaction. *UIST 1994*. p. 195-204.
11. Chatty, S. (1994). Issues and experience in designing two-handed interaction. *CHI 1994*. p. 253-354.
12. Cutler, L.D., Frohlich, B., Hanrahan, P. (1997). Two-handed direct manipulation on the responsive workbench. *I-3D*. p. 107-114.
13. Dietz, P. Leigh, D. 2001. DiamondTouch: a multi-user touch technology. *UIST '01*. p. 219-226.
14. Esenther, A., Ryall, K. (2006). Fluid DTMouse: Better mouse support for touch based interactions. *AVI 2006*. p. 112-115.
15. Flider, M. J., Bailey, B. P. (2004). An evaluation of techniques for controlling focus+context screens. *GI 2004*. p. 135-144.
16. Forlines, C., Wigdor, D., Shen, C., Balakrishnan, R. (2007). Direct-Touch vs. Mouse Input for Tabletop Displays. *CHI 2007*. p. 647.
17. Guiard, Y. (1987). Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *Journal of Motor Behavior*, 19(4), 486-517.
18. Hinckley, K., Baudisch, P., Ramos, G., Guimbretiere, F. (2005). Design and Analysis of Delimiters for Selection-Action Pen Gesture Phrases in Scriboli. *CHI 2005*. 451-460.
19. Hinckley, K., Czerwinski, M., and Sinclair, M. (1998). Interaction and modeling techniques for desktop two-handed input. *UIST 1998*. p. 49-58.
20. Hinckley, K., Pausch, R., Proffitt, D., Patten, J., and Kassell, N. (1997). Cooperative bimanual action. *CHI 1997*. p. 27-34.
21. Kabbash, P., MacKenzie, I.S. & Buxton, W. (1993). Human performance using computer input devices in the preferred and non-preferred hands. *InterCHI 1993*. p. 474-481.
22. Krueger, M., VIDEOPLACE and the interface of the future. *The art of human computer interface design*, B. Laurel, Editor. 1991, Addison Wesley: Menlo Park, CA. p. 417-422.
23. Kurtenbach, G., Fitzmaurice, G., Baudel, T., Buxton, B. (1997). The design of a GUI paradigm based on tablets, two-hands, and transparency. *CHI 1997*. p. 35-42.
24. Latulipe, C., Mann, S., Kaplan, C., Clarke, C. (2006). SymSpline: symmetric two-handed spline manipulation. *CHI 2006*. p. 349-358.
25. Leganchuk, A., Zhai, S., & Buxton, W. (1998). Manual and cognitive benefits of two-handed input: an experimental study. *ToCHI*, 5 (4). p. 326-359.
26. MacKenzie, I. S., Guiard, Y. (2001). The two-handed desktop interface: are we there yet?. *CHI 2001*. p. 351-352.
27. Matsushita, N., Ayatsuka, Y., Rekimoto, J. (2000). Dual touch: A two-handed interface for pen-based PDAs. *UIST 2000*. p. 211-212.
28. Matsushita, N., Rekimoto, J. (1997). Holo Wall: Designing a Finger, Hand, Body, and Object Sensitive Wall. *UIST 1997*. p. 209-210.
29. Maxell PenIT. <http://www.maxell.co.jp/e/products/industrial/digitalpen/>
30. Myers, B. A., Lie, K. P., Yang, B-C. (2000). Two-Handed Input Using a PDA and a Mouse. *CHI 2000*, 41-48.
31. Owen, R., Kurtenbach, G., Fitzmaurice, G., Baudel, T., and Buxton, W. (2005). When it gets more difficult, use both hands: exploring bimanual curve manipulation. *GI 2005*. p. 17-24.
32. Rekimoto, J. (2002). SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces. *CHI 2002*. p. 113-120.
33. Shoemaker, G., Gutwin, C. (2007). Supporting Multi-Point Interaction in Visual Workspaces. *CHI 2007*. p. 999-1008.
34. Ullmer, B., & Ishii, H. (1997). The metaDESK: Models and prototypes for tangible user interfaces. *UIST 1997*. p. 223-232.
35. Wu, M. and Balakrishnan, R. (2003). Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. *UIST 2003*. p. 193-202.
36. Wu, M., Shen, C., Ryall, K., Forlines, C., Balakrishnan, R. (2006). Gesture Registration, Relaxation, and Reuse for Multi-Point Direct-Touch Surfaces. *IEEE Tabletop 2006*.
37. Yee, K.-P. (2004) Two-handed interaction on a tablet display. *CHI 2004*. p. 1493-1496.