Design and Evaluation of Interaction Models for Multi-touch Mice

Hrvoje Benko[1], Shahram Izadi[2], Andrew D. Wilson[1], Xiang Cao[2], Dan Rosenfeld[3], Ken Hinckley[1]

[1]Microsoft Research
1 Microsoft Way
Redmond, WA, 98052, USA

[2]Microsoft Research Cambridge
7 JJ Thomson Avenue
Cambridge CB3 0FB, UK

[3]Microsoft Corporation
1 Microsoft Way
Redmond, WA, 98052, USA

{benko, shahrami, awilson, xiangc, danr, kenh}@microsoft.com

Abstract

Adding multi-touch sensing to the surface of a mouse has the potential to substantially increase the number of interactions available to the user. However, harnessing this increased bandwidth is challenging, since the user must perform multi-touch interactions while holding the device and using it as a regular mouse. In this paper we describe the design challenges and formalize the design space of multi-touch mice interactions. From our design space categories we synthesize four interaction models which enable the use of both multi-touch and mouse interactions on the same device. We describe the results of a controlled user experiment evaluating the performance of these models in a 2D spatial manipulation task typical of touch-based interfaces and compare them to interacting directly on a multi-touch screen and with a regular mouse. We observed that our multi-touch mouse interactions were overall slower than the chosen baselines; however, techniques providing a single focus of interaction and explicit touch activation yielded better performance and higher preferences from our participants. Our results expose the difficulties in designing multi-touch mice interactions and define the problem space for future research in making these devices effective.

Keywords: Multi-touch, mouse, surface computing, desktop computing.

Index Terms: H.5.2 [Information interfaces and presentation]: User Interfaces. – Input devices and strategies; Graphical user interfaces.

1 Introduction

In the 40 years since the early prototypes were built by Engelbart and colleagues [5], the computer mouse has become a highly optimized interaction device, allowing users to perform precise interactions with minimal effort. Recently, two separate efforts demonstrated a new possibility of adding multi-touch sensing onto the surface of the mouse, thus creating a multi-touch mouse (MT mouse). Villar et al. [24] first demonstrated the basic interaction capabilities on five hardware MT mice prototypes and Apple released the first commercially available MT mouse, called Magic Mouse [20], which allows users to perform touch gestures on top of the mouse. These MT mice (in addition to the existing multi-touch pads), present an opportunity to bring the rich multi-fingered interactions demonstrated by surface computing to the desktop environment without the cost of multi-touch screens and with all of the potential ergonomic benefits associated with mice.

However, it is unclear whether multi-touch input and the standard mouse functionality can be effectively combined as there are difficult challenges that need to be resolved in order for this vision to materialize (e.g., facilitating a firm hold of the device without inadvertent touch activation, supporting touch interactions through an indirect touch device, finding compelling interactions and use scenarios, etc.) So far, the existing work provides little support for gauging the effectiveness of multi-touch on the mouse. Villar et al. [24] outline many potential use scenarios (without supporting evaluations) and focus on implementation details of their different hardware prototypes. Magic Mouse [20] enables mostly single finger gestures with the only available multi-touch interaction being the two-finger horizontal swipe gesture.

In this paper we focus on the design problem of *how to effectively use the touch data on a mouse to manipulate on-screen objects*. To constrain the large interaction space we chose to restrict the interactions to the basic 2D manipulations: the cursor-based pointing from the desktop world and the translation, rotation, and scale manipulations of 2D objects with multiple contacts from the surface computing domain. From a user's perspective, we want to understand and identify mental models for the use of MT mouse, which can allow users to leverage the power of both touch and mouse input. While this problem may appear simple, there are many design decisions that need to be addressed. For example, are sensed touches mapped to the region around the cursor, the screen, or independently to some other object or region of interest? Is the multi-touch sensor always active, or does it need to be explicitly triggered? Do we leverage existing WIMP-based models for input focus and selection, and retrofit touch around these or do we require something completely different?

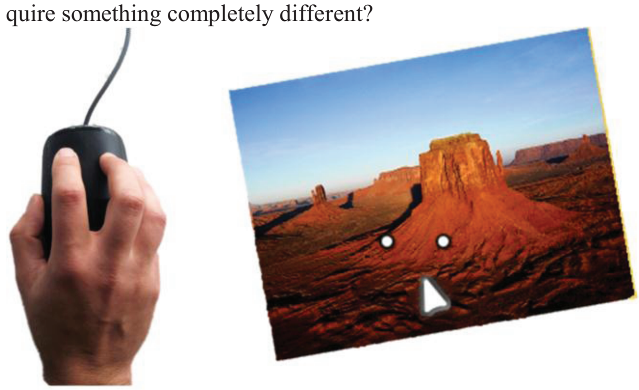

Figure 1. The multi-touch mouse device in use and the corresponding cursor and the touch points in the interface.

Our work explores the rich interaction space for MT mice and makes the following three contributions. We first describe a set of design challenges with integrating touches from the MT mouse to a desktop user interface and categorize them in a taxonomy. We then use this taxonomy to identify four techniques for integrating multi-touch with the existing cursor-based model for interaction (Figure 1). Lastly, we report on a controlled user study which compares our four techniques against two real-world baseline interactions using different devices – a multi-touch screen and a regular mouse. Our results show that MT mouse interactions were overall significantly slower than the existing baselines (by 27%); however, techniques providing a single focus of interaction and explicit touch activation yielded better performance and higher preferences from our participants. We synthesize the lessons from our observations, expose the benefits and shortcomings of our interactions given the chosen task, and highlight the directions for future research of finding the appropriate tasks and interactions in order to make these devices effective.

2 Related Work

Extending the available interaction bandwidth of the computer mouse has been the focus of numerous research projects. The most widely adopted addition to the core mouse functionality was the scroll-wheel [25]. MacKenzie et al. [16] and Fallman et al. [7] demonstrated hardware mouse prototypes which combine two mouse sensors into the single enclosure, enabling rotation sensing. Combining two mice independently, Latulipe et al. [15] explored multi-point interactions and symmetric vs. asymmetric bimanual input. Rockin'Mouse [2] added a tilt sensor to the mouse, which allowed for control of 4 degrees of freedom and was primarily used for manipulation of virtual 3D environments.

While the standard mouse operation requires a flat surface for interaction, researchers experimented with devices featuring accelerometers and gyroscopes (e.g., [1][9]), allowing the user to operate the mouse while holding it in mid-air. Extending this idea, VideoMouse [11] added a camera inside the mouse case pointing down at a custom 2D grid pattern printed on a mouse pad, which supported 6DOF input. Cechanowicz et al. [3] explored the use of pressure-augmented mice, where one or more pressure sensors mounted on the mouse control the cursor position as well as multiple levels of discrete selection. Recently, Kim et al. [14] investigated the inflatable mouse concept and mentioned one of its potential uses as a pressure sensitive input device.

Our work is closely related to the projects which explore integrating another spatial sensor on top of the mouse device, which itself is already a relative spatial sensor. The PadMouse project [3] added a single touch pad to the mouse and demonstrated the benefits of such a configuration for activating commands and modifiers using finger gestures. Others incorporated a relative-position sensing mini joystick (e.g. TrackPoint Mouse [19]) or a miniature trackball (e.g. Apple's Mighty Mouse [20]) on top of a mouse. Hinckley et al. [11] provide a comprehensive overview of the touch-sensing input devices and demonstrate the Scrolling TouchMouse prototype which can sense when the user is in contact with the device and enable additional modes in the interface (e.g., when not touching the mouse, the toolbars in the interface disappear). Taylor and Bove [22] explored the concept of classifying the user's grasp and adapting the device interface accordingly. While related, these prototypes do not attempt to track user's touches or enable multi-touch interactions on an input device.

Figure 2. The Cap Mouse prototype (developed by [24]) used in our investigation: the labels mark the curved capacitive multi-touch sensor area and the inactive grip zone that can be used for gripping the mouse without touching the sensor.

We base our explorations on the Cap Mouse prototype designed by Villar et al. [24] (Figure 2) which uses a capacitive multi-touch sensor integrated into the curved top of the device. The only other components of the Cap Mouse prototype are hidden in its base: the standard mouse position sensor and one click button which is activated by pressing on the entire front of the device.

In the surface computing domain, several projects explored multi-touch capacitive sensing (e.g., DiamondTouch [6], SmartSkin [21]) and approximating mouse input given multi-touch input (e.g., DTMouse [8] and SDMouse [18]). The benefits of multi-touch interactions can also be achieved without the direct onscreen interactions (e.g., iGesture pad [13] or the Apple MacBook multi-touchpad [20]). Malik et al. [17] explored a camera-based multi-touchpad system designed as an indirect input system to a large relatively distant display. They suggested mapping the user's interactions onto a movable region of interest on the large display.

Moscovich et al. [19] explored the concept of area cursors defined by multiple touch points on a multi-touch pad, and suggested interaction techniques that aggregate the behavior of multiple touches into the behavior of a single on screen cursor. Their work, while not based on the position sensing device like a mouse, but rather on a stationary multi-touch pad, is closely related to ours as they explore the remote mapping of touches to the interface. However, while they offer three interesting interaction techniques, they do not provide any formal comparisons between them and there is little information about the users' preferences or performance with these techniques.

3 The Design Space of MT Mouse Interactions

In this section we characterize the problem of integrating multi-touch interactions into an existing cursor-based desktop environment. The key interaction issue with MT mice is that the user is not generating one but *two* continuous input streams (mouse and touch input), which both need to be processed and used to interact with and manipulate on-screen content. It is possible to consider these input streams independently where touch sensing can be used for gestures which are mostly independent of mouse cursor location (e.g., Magic Mouse [20]). However, we focus on interactions which combine these two input streams, i.e., those which use touches in addition to the cursor for manipulations in the interface. In doing so, we define a taxonomy which describes various design tradeoffs and considerations that need to be considered when designing multi-touch interactions on a mouse.

Our taxonomy includes four core dimensions: *mapping*, *activation*, *focus*, and *feedback* (Figure 3). In describing these, we highlight one of the key tensions in determining our interaction model: when to defer to a traditional mouse-based cursor model, when to leverage a multi-touch model or when to create a hybrid of both. Our hope is that this taxonomy will help the reader understand the complexity of available design options, and provide insights in defining a coherent interaction model for users.

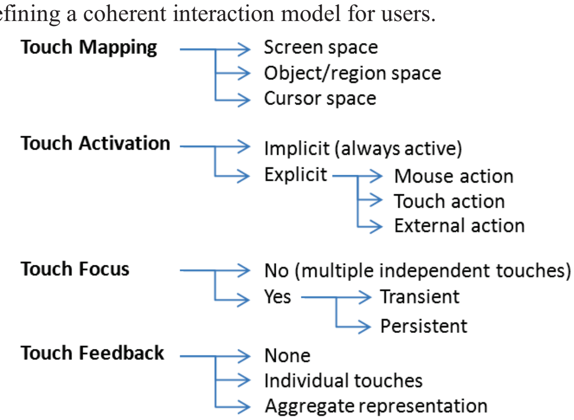

Figure 3.The design space taxonomy of MT mouse interactions.

3.1 Touch Mapping

MT mouse (and any other mouse) is an indirect interaction device, where the input and the output are spatially decoupled. Furthermore, the MT mouse has a smaller touch sensor area than the screen's output, making it necessary to decide how to *map* the touch positions from the sensor onto the interface. We identify three key ways of mapping the touches from the sensor onto the interface: *screen, object/region,* and *cursor mapping.*

Screen space mapping transforms the sensor data to the full bounds of the screen (e.g., touching the top left of the MT mouse sensor maps the touch to a location at top left point of the screen). This mapping suffers when there is high mismatch between input and output size and resolution since small movement on the sensor can result in large jumps on the screen.

Object/region space mapping bounds the touch to a specific on-screen region. Such a region can be defined by an on-screen object (e.g., touches can be mapped around the center of the object and might be bound by the object's shape). This could also provide an arbitrary mapping depending on the position and size of the object/region.

Cursor space mapping bounds the touch to a predefined or dynamic area centered around the mouse cursor (as suggested by [19], [24]). The position of the touch points can dynamically change dependent on the position of the cursor.

Note that all these mappings can be considered *absolute*. i.e., when the user touches the center of the mouse sensor, a touch is registered in the center of the bounds whether those are of the screen, object/region or cursor. While this seems the most appropriate model for MT mice, touchpads can also work in a relative mode, allowing the mapped touch point to effectively be *clutched*. While this is normal for single touchpad operation, the ability to support clutching of touch points is problematic on an MT mouse. To support clutching, one would essentially require each finger to be assigned a persistent touch point, i.e., a cursor as suggested in [19]. There is a correspondence problem in determining which sensed finger corresponds to which touch point, since the geometric configuration of the cursors may also quickly become inconsistent with that of the user's fingers, caused by clutching and imprecise mapping.

3.2 Touch Activation

The second category in our taxonomy is the notion of touch activation. This refers to the act that enables the data from the MT sensor to be active in the interface. We state that the activation can be either *implicit* or *explicit.*

The *implicit* mode has no separate activation and the touches are active as soon as they land on the MT mouse. This, in principle, is similar to the default behavior of the touch screens, which support only a two-state interaction model (off – on).

In the *explicit* mode, touches are not active by default, but require some specific action in order to be activated. We group the possible actions into *mouse actions* (e.g., mouse clicks or mouse dwell), *touch actions* (e.g., taps or touch movement), or some *external actions* (e.g., key press). The explicit mode is closely related to the standard three-state mouse interaction model (first explained by Buxton [4]) which enables the cursor to remain in the inactive hover mode until the user is ready to engage by pressing the mouse button. Enabling the hover state allows us to preview where the touch will occur before committing the touch. Explicit activation can also be beneficial for suppressing accidental touches. For example, one needs to maintain a grip of MT mouse to carry out pointer-based manipulations. So even if it is not the user's intention to trigger multi-touch interactions, there may be sensed multi-touch data that could trigger false interaction.

3.3 Touch Focus

In addition to mapping touch positions onto the interface and activating touches, there are several options in choosing the on-screen object(s) to interact with. In a desktop environment, this is usually referred to as *focus,* i.e. *choosing a single object in the interface to receive input exclusively*. However, this notion of focus contrasts with the interaction model of direct multi-touch interfaces, where there is no single focus model, and instead multiple touches may interact with multiple objects simultaneously. Occupying a middle ground between conventional desktop interface and direct multi-touch interface, MT mouse interactions need to choose to *have a focus or not*.

If the focus model is not chosen, each touch behaves independently and simultaneous actions on multiple objects are possible. Alternatively, one can think of not having a specific focus model as having the ability to have *multi-foci* interactions*.* However, if the focus model is chosen, only a single object receives all the touch input, which leads to the choice of how to decide which object is in focus. This choice is often closely coupled with the activation action, since it usually makes sense to use the same action to both select an object and activate the touches. There are two main selection mechanisms we identified from desktop interfaces: *transient selection* and *persistent selection*.

With *transient selection* of focus, the object maintains its focus only while a selection event is happening (e.g., while the cursor is above the object, while the user is clicking on an object, or while the touches are moving over an object).

With *persistent selection,* once selected the object remains in focus until some other action deactivates it. The persistent mode is in effect a toggle state where touches are activated until some other event deactivates them (e.g., touches are active while the object remains selected, or touches are activated and deactivated by mouse click). WIMP interfaces primarily use the persistent selection method for cursor interactions.

3.4 Touch Feedback

The MT mouse's inability to directly interact with the interface as with multi-touch screens or surfaces, means that the user will lose the natural visual feedback of the input from their hands touching or hovering above the display, and in turn requires some on-screen feedback to mark the location of their touches.

There are three feedback categories available for visualizing user's touches: *no explicit feedback*, *individual touch feedback*, and *aggregate touch feedback.* When there is no explicit touch feedback, the user is left to deduce her actions from the resulting manifestation of the objects in the interface (e.g., the object's movement). Alternatively, feedback visualization can include each individual touch. The example of this category is the Villar et al.'s MT Cloud visualization [24] (seen in Figure 1 and Figure 4). Lastly, feedback can also be presented in an abstract form, possibly aggregating multiple touches into a single representation (e.g., the cursor itself could change appearance based on the number and position of the touches). These techniques of course may be combined to give the richest feedback, possibly at the risk of over-complicating the interaction.

4 MT Mouse Interaction Techniques

We have identified a multidimensional design space, with many aspects and combinations to explore. It is beyond the scope of this paper to formally evaluate the entire space, and doing so is probably not worthwhile since not all combinations make for good interaction methods.

We instead chose to consider two key aspects of the design space, critical in defining an interaction model for MT mouse – *focus* and *activation*. Specifically, we would like to answer two questions:

1) What kind of focus model should MT mouse interactions have?
2) Does it make sense to have an explicit activation mechanism for MT mouse interactions?

Based on these questions, we designed four promising MT mouse interaction techniques to evaluate these tradeoffs and collect user preferences.

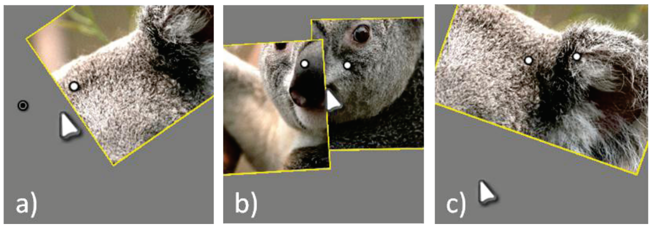

Figure 4. Touch feedback visualization shows the position of the user's fingers on the sensor: (a) with a no-focus technique (e.g., IT) it is possible to have one touch be active (white) and the other inactive (black), or (b) simultaneously act on multiple objects; (c) with persistent focus, the touches can remain mapped to the object while the cursor can move independently.

4.1 MT Mouse Independent Touches (IT)

The *Independent Touches* technique is based on a cursor space mapping and is most similar to the default behavior of a multi-touch screen. There is no notion of a single object in focus. Every object responds to touch points that are directly over them, therefore allowing for simultaneous multi-object manipulation (Figure 4b), and this technique has an implicit activation mode where touches are active on contact without any additional activation action.

4.2 MT Mouse Hover Cursor (HC)

To investigate the difference between having a focus or not, we designed the second technique, *Hover Cursor,* which differs from the Independent Touches technique by having a transient focus determined purely by the location of the cursor in the interface, i.e., only the object directly under the cursor responds to all present touch points, regardless of whether they are over it or not. The activation is implicit as no explicit action is needed to activate the touches (beyond the actual contact with the sensor).

4.3 MT Mouse Click 'n' Hold (CH)

This third technique, *Click 'n' Hold*, has a transient focus similar to Hover Cursor, but requires an explicit activation triggered by the mouse button (i.e., the touches are active only while the user holds the mouse pressed, and the touches affect only the object under the cursor).

4.4 MT Mouse Click Selection (CS)

To explore the difference between the transient and the persistent focus models, we designed the *Click Selection* technique, which has a persistent input focus model. In order to activate the selection, the user clicks on an object of interest and the object remains in focus until de-selected. The touches are then mapped using object/region mapping to the selected object and can be completely decoupled from the cursor (Figure 4c).

All four MT mouse techniques are summarized in Table 1. To limit the scope of this work, we decided to keep the other design options from our taxonomy fixed. For consistent feedback we used Villar et al.'s MT Cloud visualization [24] with one small modification: each touch contact was shown as active or inactive by coloring the touch "bubble" white or black respectively (Figure 4a).

5 User Evaluation

We conducted a controlled user experiment to better understand how design choices of touch activation and focus impact the usability of the multi-touch manipulation interactions on the MT mouse. The user feedback from the previous pilot experiment [24], which informally compared different MT mouse hardware devices, hinted at the numerous tradeoffs and possibilities with using the multi-touch interactions on the mouse; however, such tradeoffs were never evaluated. In this study, we tested multiple MT mouse interaction techniques against each other and against two existing baseline interactions (multi-touch screen and regular three button mouse) to provide deeper insight into this complex design space. We were interested in both the subjective preference of our participants and the quantitative performance data.

5.1 Method

5.1.1 Conditions

We tested four MT mouse techniques which varied in focus and activation models (mnemonically labeled IT, HC, CH, and CS in Table 1 and in all subsequent figures). For all MT mouse techniques we chose to keep the Villar et al.'s MT Cloud [24] feedback available and each touch cursor was shown either in white (currently active) or black (currently inactive). Inactive cursors did not affect any objects in the environment.

Cond.	Technique Name	Focus	Activation
IT	MT Mouse Independent Touches	No	Implicit
HC	MT Mouse Hover Cursor	Yes - Transient	Implicit
CH	MT Mouse Click 'n' Hold	Yes - Transient	Explicit
CS	MT Mouse Click Selection	Yes - Persistent	Explicit
M	Regular 3-button Mouse	Yes - Transient	Explicit
T	Multi-touch Screen	No	Implicit

Table 1. Summary of all the conditions tested in our experiment together with their defining focus and activation models. Baseline conditions Mouse (M) and Touchscreen (T) shown in green.

For comparison, we selected two existing input technologies available on the desktop today as our baseline conditions: multi-touch screen and regular three button mouse. In Touchscreen (T) condition the participant could directly touch the multi-touch screen with multiple fingers to move, rotate and scale the onscreen object. In the Mouse (M) condition participants used the left mouse button of the regular mouse to move the object, the right mouse button to rotate the object around its center and the scroll wheel to scale the object. The Mouse condition was the only one where translation, rotation and scale were completely separated and activated with different buttons rather than touches. As such, it provides more independent control to the user and it does not fall into the category of multi-touch style interactions, which makes it difficult to draw direct comparisons to MT mouse techniques. However, we felt that it is important to include it in the study as a baseline given the extensive use of mouse device in today's computing and the fact that we attempting to augment the mouse with the multi-touch sensor. Table 1 summarizes our experimental conditions.

5.1.2 Tasks

The users were asked to perform two tasks: the puzzle task (for practice) and the docking task (for testing). We chose tasks that required the user to perform 2D spatial manipulations typical of touch-based interfaces (e.g., translating, rotating and scaling a photo, or manipulating a map). The puzzle assembly task was used as a practice task before each condition to familiarize the participant with the device and the condition tested. The participants were asked to assemble a four piece puzzle where each piece needed to be moved, scaled and rotated into place in order to assemble the final image (Figure 5a). Participants spent approximately 5 minutes performing this task for each condition.

The second task was a docking task, which required the participant to find the test object (labeled "1") hidden under two "distracter" objects, move the test object to the target location and rotate and scale it to match the target (Figure 5b,c,d). The distracter objects were chosen to provide the opportunity for the user to manipulate several objects at once if so desired and if the technique supported it (e.g., Touchscreen or Independent Touches). The distance between the initial object location and the target location was fixed (860 pixels), but rotation varied (60 degrees clockwise and 60 degrees counterclockwise) as well as scale (20% smaller and 20% larger). The trial ended automatically when the test object was released and was roughly aligned with the target (with the tolerance of 15 pixels in position, 5 degrees in rotation and 10% in scale). Rather than demanding pixel precise docking, the docking tolerance was chosen to evaluate the participant's ability and their speed to select and spatially manipulate the on-screen object to their liking, rather than their ability to precisely position the object.

In addition to the practice puzzle assembly task described above, participants were asked to complete 8 practice trials with the docking task before running the actual test to ensure that they were comfortable and able to perform the task.

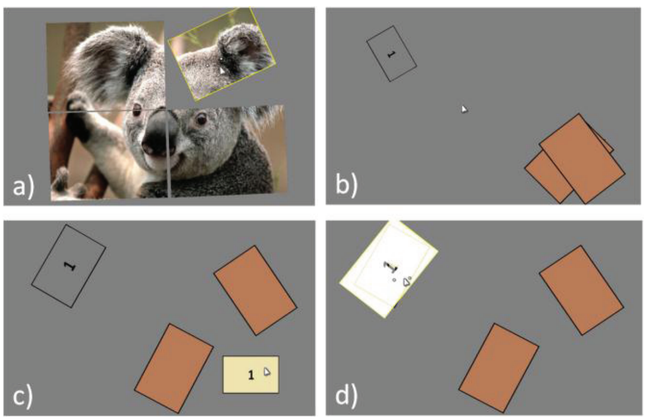

Figure 5. Two tasks in our experiment: (a) a 4 piece puzzle task (practice) and (b–d) the docking task (test) where the subjects (c) first uncovered the test object labeled "1" and (d) placed this object at the target location at the correct scale and orientation.

5.1.3 Measures

Our test application recorded the running time of each trial, the number of finger touches and mouse clicks during the trial, and the docking errors (in position, rotation and scale) from the given target location. In addition, we recorded video of the participants' hands using the MT mouse. This allowed us to observe the small variations in the users grip and hand pose.

5.1.4 Participants

We recruited 12 paid participants (6 women), all right-handed, between the ages of 21 and 41 (mean age of 29) from the local population. All were expert computer users with 6 using the mouse as their primary pointing device and 6 using the touchpad. Seven participants had some experience with multi-touch devices (e.g., iPhone, MacBook's multi-touch pad, or Microsoft Surface) and none had seen or interacted with a MT mouse.

5.1.5 Design

We employed a within subjects design with six main conditions: Independent Touches, Hover Cursor, Click 'n' Hold, Click Selection, Mouse, and Touchscreen. For the MT mouse conditions we counterbalanced for ordering effects across participants using a Latin square. The two baseline conditions (Mouse and Touchscreen) were always grouped together, randomly ordered, and completed first by half of our participants and last by the other half. For each condition we tested 1 distance x 2 rotations x 2 scales x 4 repetitions (16 trials total). Given 6 conditions, we recorded 96 trials per participant. The entire session took 90 min.

5.1.6 Apparatus

We performed our tests on a Dell XT2 Tablet PC which has a multi-touch sensitive screen (1280x800 pixels, 12.1 in diagonal). For MT mouse conditions we used the Cap Mouse multi-touch mouse prototype presented in [24]. The Cap Mouse prototype has a capacitive-sensing grid overlaid on its surface; the capacitive sensor is capable of sensing multiple finger contact points.

Following the feedback from our pilot users, we added an "inactive grip zone" on each side of the mouse to enable the user to grip and click the mouse without registering any contacts (Figure 2). Touch sensing was disabled in those areas and they were marked with tape for easy tactile disambiguation. For the Mouse condition we used the Microsoft Comfort Optical Mouse 1000. The laptop's multi-touch screen was used for Touchscreen condition. By performing all conditions on the same computer, we controlled the display size as well as visible target sizes.

5.1.7 Procedure

Participants were first given a brief introduction to the study and the apparatus after which each participant completed practice and testing trials for each of the 6 conditions. After completing each condition, we asked the participant to complete a short questionnaire stating their preferences with respect to the condition. At the end of the study, the participant was asked to rank the conditions according to their preference and the ease of use as well as to provide feedback on their experience.

5.1.8 Data Selection

To avoid the influence of outliers and the typical skewing associated with response time data, for each participant and in each Condition x Rotation x Scale combination, we performed our analysis on the median task completion times. Since our participants performed 4 repetitions for each case, selecting the median value removed the influence of the fastest and the slowest trial in each case and took the average of the remaining 2 trials.

5.1.9 Hypotheses

(H1) Touchscreen should be the fastest overall as it is based on simple direct physical manipulation and because it allows the user to simultaneously perform translation, rotation, and scale, eliminating the need for sequential actions.

(H2) Click Selection should be the fastest and the most preferred MT mouse condition because it uses persistent focus model, clearly separating mouse and touch control, and it uses an explicit activation model which does not require the user to keep the mouse pressed while interacting.

(H3) Independent Touches should be the slowest overall as it has no focus and no explicit activation and therefore it should be the most prone to accidental actions.

(H4) Participants with prior experience with multi-touch devices should exhibit better performance with the MT mouse.

5.2 Findings

5.2.1 Performance Results

We performed the 6 (Condition) x 2 (Rotation) x 2 (Scale) repeated measures ANOVA on the within-subjects effects of the task completion time. We found no significant effects with Rotation or Scale factors. This was consistent with our expectation that rotating clockwise/counterclockwise or scaling up/down would not produce a different effect on the overall task completion time given their completely symmetric operations. However, we found significant effects with Condition ($F_{(5,35)}$=20.236 $p<0.001$, Figure 6). Further pairwise analysis (with Bonferroni correction) revealed that the differences between Independent Touches and Hover Cursor, Click 'n' Hold and Click Selection, and Mouse and Touchscreen were not significant. The differences between these three groups (i.e., IT&HC vs. CH&CS vs. M&T) were significant. This suggests that the lack of an explicit activation model for Independent Touches and Hover Cursor hurt the performance of our participants.

While these results do not conclusively confirm H1, H2, or H3, we at least observe the trends predicted by H1 and H3. Mouse (7252ms) was overall the fastest, followed by Touchscreen (7459ms), while the fastest MT mouse conditions were Click 'n' Hold (9217ms) followed by Click Selection (9782ms). The slowest condition overall was Independent Touches, taking on average more than 12 seconds to complete a trial. The performance time difference between the fastest MT mouse condition Click 'n' Hold and the fastest baseline Mouse was roughly 2 seconds (27%).

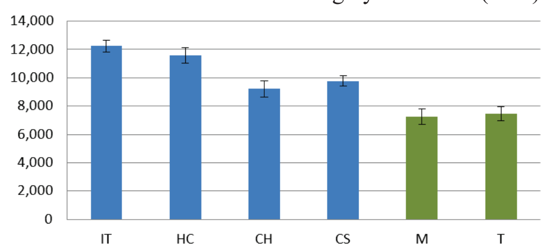

Figure 6. Completion times across Condition (milliseconds +/- SEM).

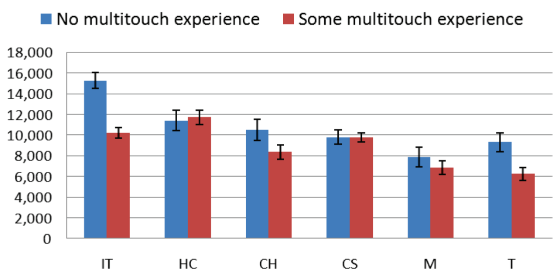

Figure 7. Interaction of Condition and Experience on completion times (milliseconds+/-SEM).

For a better understanding of the Condition effects on task completion time, we examined the between-subjects factors of Gender and Experience. In our experiment half the participants were female and 7 out of 12 had some prior multi-touch experience (e.g., used the iPhone, MacBook's multi-touch pad, or Microsoft Surface). We found no significant effects of Gender, but the interaction of Condition and Experience yielded significant results ($F_{(5,35)}$=3.069,p=0.021) seen in Figure 7. This implies that participants with prior multi-touch experience performed significantly better with conditions Independent Touches and Touchscreen conditions, but not with other conditions (Click 'n' Hold had borderline significance).

This did not confirm H4, since not all MT mouse conditions benefited from prior multi-touch experience. While it is not surprising that prior multi-touch experience helped with Touchscreen and not with Mouse, it is less obvious why it made a difference for some MT mouse techniques and not for others. We postulate that Independent Touches is the MT mouse technique most similar to the default behavior of the multi-touch screen (since the touches have implicit activation and there is no focus model) and therefore that is also the condition where we observed the biggest difference between experienced and inexperienced multi-touch users (>5 seconds). Conversely, Click Selection is most similar to the standard mouse behavior: touches are active only on selection and movement is completely separate from rotation and scaling. Therefore it is foreseeable that prior multi-touch experience does not seem to make as much of a difference for Click Selection.

5.2.2 Touch and Click Analysis

We further analyzed the way each condition was performed in our experiment (Figure 8). Specifically, we analyzed the number of new contacts and the number of mouse clicks during the each trial as an indication of how much effort was expended. These measures are approximate at best since our participants were not instructed to minimize the number of touches or clicks. However, these results shed light on how each condition was actually performed. For example, more contacts may indicate that the user had to reposition their touches more frequently.

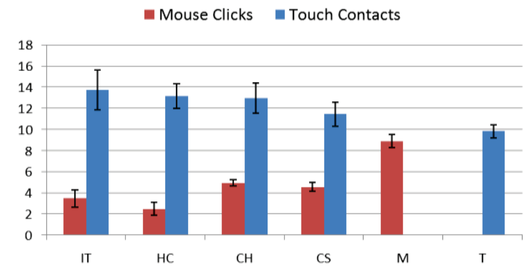

Figure 8. The average counts (+/- SEM) of mouse clicks and touch contacts across conditions. Note: Mouse had no contacts and Touchscreen had no mouse clicks.

A repeated measures ANOVA on the contact data found no significant effects on the Condition. In summary, participants did not use significantly different number of contacts per trial. Analysis of clicks revealed an interesting finding that our participants clicked even when not required when using Independent Touches and Hover Cursor. In those techniques, clicking was not specifically disabled (as that is not mechanically possible on our device), but all of the interactions could be completed without a single mouse click and participants were instructed to do so. However, several noted that clicking "just felt more natural." As expected, Mouse required at least twice as many clicks as MT mouse conditions, since the actions had to be completed sequentially by clicking multiple buttons.

5.2.3 Subjective Feedback

Participants filled out a post-experiment questionnaire rating their experience with the six conditions on a 7 point Likert scale (1 being most negative and 7 being most positive). They were asked to comment on the following categories: ease of use, control over their actions, perceived speed, amount of accidental activation, and helpfulness of touch feedback.

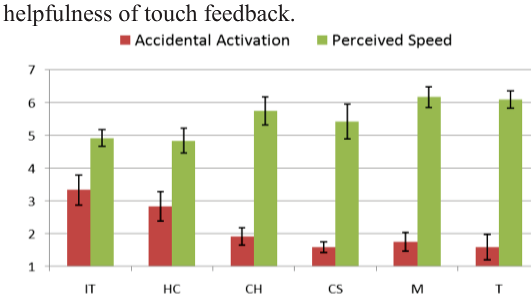

Figure 9. Aggregated subjective responses (+/- SEM) for questions "Did you accidentally activate objects you did not intend to?" (1–Never, 7–All the time) and "I feel that I was fast at accomplishing this task using this technique." (1–Strongly disagree, 7–Strongly agree).

The participants' responses on perceived speed closely correlate with their actual performance data, with Mouse and Touchscreen conditions ranked as the fastest, followed by Click 'n' Hold and Click Selection, and then Independent Touches and Hover Cursor (Figure 9). To explain the differences between observed task times for the MT mouse conditions it is instructive to look at responses to the question “Did you accidentally activate objects you did not intend to?” (1 – Never, 7 – All the time). Aggregated responses indicate that participants noticed a higher number of errors with Independent Touches and Hover Cursor conditions (Figure 9).

When asked to select their preferred MT mouse condition, Click ‘n’ Hold was rated top by 4 out of 12 participants, followed by Click Selection and Hover Cursor (preferred by 3 participants each). The participants also ranked all 6 conditions according to preference. Touchscreen was highest rated with aggregate ranking scores of 4.5 respectively (score of 6 = top rating) (Figure 10). Click ‘n’ Hold was the highest ranked MT mouse technique (aggregate score of 3.833). While no MT mouse condition came out on top overall, 7 out of 12 participants preferred one of the MT mouse conditions to the regular mouse and 5 out of 12 preferred an MT mouse condition to Touchscreen. As expected, 3 participants explicitly commented about hand fatigue experienced in condition Click ‘n’ Hold. However, 2 of those same participants noted that they felt faster and more precise with Click ‘n’ Hold than with other MT mouse conditions.

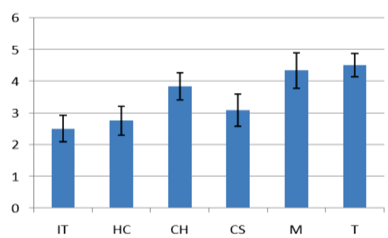

Figure 10. Aggregated preference ratings for all 6 conditions (+/- SEM) (1 - bottom, 6 - top ranking).

5.2.4 Finger and Grip Analysis

We have also observed a convergence on the hand pose used with the MT mouse conditions. Nine out of 12 participants used the standard “mouse” grip, with their index and middle fingers used for multi-touch interactions and the thumb and ring fingers for maintaining the grip of the device (Figure 11a). While consistent with the use of the regular mouse, this pose places significant limits on the extent of the finger separation. When asked about their preferred pose, a participant said “it simply felt better to reserve my index and middle finger for multi-touch and other fingers for clicking with the mouse.”

Figure 11. Different MT mouse hand poses observed in our experiment: a) the “mouse” grip – 2 interactive fingers (9 participants), b) the “pinch” grip – 2 interactive fingers (2 part.) and c) the “modified mouse” grip – 3 interactive fingers (1 part.).

Two participants used the “pinch” grip (thumb + index fingers) for interactions (Figure 11b). This allowed the use of the comparatively dexterous thumb, but also required that in the Click ‘n’ Hold condition they click the mouse with the same fingers used for interacting. Finally, one participant used the middle three fingers to perform multi-touch interactions (Figure 11c). This participant also tended to hold the middle finger stationary, while performing rotation and scaling manipulations by moving the ring and the index fingers around it.

We also expected that the pinch posture would be the dominant finger combination for use in Touchscreen condition; however, we observed that regardless of condition ordering, 6 out of 12 participants used their index and middle fingers, and two additional participants frequently switched between thumb + index and index + middle combinations.

6 Discussion

Our user experiment results reveal several important findings that have implications for the design of the MT mice interactions. Some of our results are negative, since no MT mouse condition performed better than the existing baselines, as their overall performance was about 27% slower than Mouse (for Click ‘n’ Hold). It is possible that this is due to the difference in experience that our participants have with a regular mouse and the complete lack of experience with an MT mouse. While no short term learning effects were observed, it would be interesting to observe how user performance improves with longer use of the MT mouse.

Furthermore, compared with Touchscreen, there are many scenarios where including multi-touch interactions in an indirect pointing device would be advantageous and yield improved performance compared to the direct touch. For example, Malik et al. [17] explored the use of a large distant display where distance and fatigue clearly penalizes direct touch interactions. By testing our conditions on a relatively small screen we did not highlight all the benefits of our device. Rather, our current comparison is a more realistic everyday use scenario. Participants also noted that “vertical touchscreens are tiring” and that they liked MT mouse because it allowed them to interact using a method similar to that of a touchscreen, but with “little effort and in a horizontal plane”.

In hindsight, it is also possible that the chosen task gave an unfair advantage to our baselines (Touchscreen and Mouse). The constraints on finger and wrist movement when using the MT mouse do not lend themselves to larger motions potentially required of larger rotations or scales. This might have yielded repeated clutching in MT mouse conditions which was not present with the baselines, but could have been addressed by introducing a gain function. Also, the chosen docking task was easy to accomplish using the separable degrees of freedom on the mouse given the strategy of centering the object on target, then rotating, and then scaling. Latulipe et al.’s [15] Image Alignment Task is difficult to perform with a regular mouse, and so might be an interesting task to test with MT mouse.

Lastly, it is possible that the MT mice might be best utilized for multi-touch gestural input mostly independent from the current cursor position, as is the case with the limited set of Magic Mouse gestures [20].

6.1 Need for Focus and Explicit Touch Activation

We now discuss the differences between MT mice conditions, the primary focus of our study. Our observations suggest that the users were faster and showed greater preference for those techniques which focused their touches on a single object and which used an explicit activation model (Click ‘n’ Hold and Click Selection). While not conclusively, Click ‘n’ Hold was the fastest and the most preferred MT mouse technique. The higher number of perceived errors in Independent Touches and Hover Cursor might be a consequence of the lack of explicit activation and lack of the “hover” state to preview actions.

This finding suggests that MT mice require an interaction model similar to the three-state model available on the mouse [4] (i.e., off, hover, and on states), rather than the two state model of a direct touchscreen (i.e., off and on states only). Not providing an explicit activation mechanism works fairly well for direct touch screens where the user can easily predict where their touches will land. However, since MT mouse is a device with an indirect mapping to the screen, the user cannot easily predict where their touches will appear. It is for this reason that supporting a preview state (i.e., hover) and explicit activation is so critical.

6.2 Enhanced Mouse, Not a Touchscreen Substitute

While it is possible to think of MT mouse as a substitute for direct touch interactions in the interface, it is more appropriate to think of it as an enhanced mouse device. The need for the mouse-like hover model is just one reason, but all our observations suggest that users assume this device to have mouse-like properties. First, we found that our study participants clicked even when not required. Second, we observed that they preferred to grip the device like a regular mouse even if they then could not user their thumb. Finally, although we deliberately designed the task to include two distractor objects to provide an incentive for moving both at once, we observed that no participant *ever* attempted to move two objects simultaneously in the Independent Touches condition even though this action was specifically demonstrated to them. Rather, this behavior was observed only in error, i.e., when the participant missed the object they were trying to interact with. This suggests that the single focus model (i.e., mouse cursor model) is probably sufficient for MT mouse use.

6.3 Picking the Mental Model and the Appropriate Feedback

While it is possible and sometimes beneficial to think of touches and the mouse cursor as separate input streams, one of our study participants suggested a mental model which considered them part of the same input stream - that of his hand. He pointed out that he thought about the interactions in anthropomorphic ways as "my hand is the cursor and touch bubbles are my fingers". Consequently, he also complained that technique Click Selection actually "breaks" this model since the "fingers" can now be disembodied from the "hand". Other participants might have not noticed this and commented that "it felt natural [with condition Click Selection] to control rotation/size independently of position."

These comments highlight the importance of establishing a clear, simple and memorable model for the user. Performing object manipulations in a manner similar to the real-world physical analogue provided a simpler overall interaction model than the one needed for the regular mouse, for which one participant observed that the "regular mouse was harder than MT mouse because I had to remember which control each mouse button corresponds to." We think that the hand + fingers mental model very clearly addresses how one might think of an MT mouse device, while it is also consistent with the goal of enabling touch screen-like interactions on a mouse. However, even with such a simple mental model, all of the issues exhibited in our taxonomy remain. For example, the aforementioned "break" in the mental model hints that the touch feedback (in its current form) might be confusing in some cases. We speculate that some of the interactions might be as easily performed without the touch feedback. While we have no way of confirming this from our data, the subjective preferences hint at this potential, since people found the individual touch feedback was less useful for MT mouse conditions other than Independent Touches. This should not be surprising, since in those conditions the exact position of the touches did not matter as the aggregate action was affecting either the object underneath the cursor or the previously selected object. It remains future work to investigate whether providing explicit visualizations of touch positions has an impact on the performance or preference of the MT mouse.

7 Conclusion

This paper presents the first effort at defining basic interaction challenges for touch-based 2D manipulation interactions using a MT mouse. We described and evaluated four interaction techniques that facilitate multi-touch interactions on these devices, each differing in terms of focus and activation methods. While the MT mouse interactions' performance on our chosen task lagged compared to the baseline methods, we have identified many important aspects that need to be addressed for effective MT mouse use such as the importance of single focus and explicit touch activation mechanism. With commercial availability MT mice, it is important to further investigate their usage models, explore the novel interactions they enable, and highlight compelling use scenarios. We hope that our work sets the useful design framework for this future research.

References

[1] Air Mouse. Logitech. http://www.logitech.com. (Sept. 2009).

[2] Balakrishnan, R., Baudel, T., Kurtenbach, G., and Fitzmaurice, G. (1997). The Rockin'Mouse: Integral 3D Manipulation on a Plane. *In Proc. ACM CHI '97*. 311–318.

[3] Balakrishnan, R. and Patel, P. (1998). The PadMouse: Facilitating Selection and Spatial Positioning for the Non-Dominant Hand. *In Proc. ACM CHI '98.* 9–16.

[4] Buxton, W. (1990). A Three-State Model of Graphical Input. *In Proc. Human-Computer Interaction - INTERACT '90*. 449–456.

[5] Cechanowicz, J., Irani, P., and Subramanian, S. (2007). Augmenting the Mouse with Pressure Sensitive Input. *In Proc. ACM CHI '07.* 1385–1394.

[6] Dietz, P. and Leigh, D. (2001). DiamondTouch: A Multi-User Touch Technology. *In Proc. ACM UIST '01.* 219–226.

[7] Engelbart, D. C. et al. (1968) A Research Center for Augmenting Human Intellect. Stanford Research Institute, Menlo Park, CA.

[8] Esenther, A. and Ryall, K. (2006). Fluid DTMouse: better mouse support for touch-based interactions. *In Proc. AVI '06.* 112-115.

[9] Fallman, D. and Yttergren, B. (2007). The Design of a Computer Mouse Providing Three Degrees of Freedom, *HCI International '07.*

[10] Go Pro Air Mouse. Gyration. http://www.gyration.com (Sept. 2009)

[11] Hinckley, K. and Sinclair, M. (1999). Touch-sensing input devices. *In Proc. ACM CHI '99.* ACM Press. 223–230.

[12] Hinckley, K., Sinclair, M., Hanson, E., Szeliski, R., Conway, M. (1999). The VideoMouse: A Camera-Based Multi-Degree-of-Freedom Input Device. *In Proc. ACM UIST '99.* 103–112.

[13] iGesture Pad. (2009). FingerWorks. http://www.fingerworks. com. (Sept. 2009).

[14] Kim, S., Kim, H., Lee, B., Nam, T.-J., and Lee, W. (2008). Inflatable Mouse: Volume-Adjustable Mouse with Air-Pressure-Sensitive Input and Haptic Feedback. *In Proc. ACM CHI '08.* 211–224.

[15] Latulipe, C., Kaplan, C.S., and Clarke, C.L.A. (2005). Bimanual and Unimanual Image Alignment: an Evaluation of Mouse-based Techniques. *In Proc. ACM UIST '05.* 123–131.

[16] MacKenzie, I. S., Soukoreff, R. W., and Pal, C. (1997). A Two-Ball Mouse Affords Three Degrees of Freedom. *ACM CHI Extended Abstracts '97.* 303–304.

[17] Malik, S., Ranjan, A., and Balakrishnan, R. (2005). Interacting with Large Displays from a Distance with Vision-tracked Multi-finger Gestural Input. *In Proc. ACM UIST '05.* 43–52.

[18] Matejka, J., Grossman, T., Lo, J., and Fitzmaurice, G. (2009). The design and evaluation of multi-finger mouse emulation techniques. *In Proc. ACM CHI '09.* 1073-1082.

[19] Moscovich, T. Hughes, J.F. (2006). Multi-finger cursor techniques. *In Proc. Graphic Interface '06.* 1–7.

[20] Magic Mouse. Apple. http://www.apple.com/. (Dec. 2009).

[21] Rekimoto, J. (2002). SmartSkin: an Infrastructure for Freehand Manipulation on Interactive Surfaces. *In Proc. ACM CHI '02.* 113-120.

[22] Taylor, B. T. and Bove, V. M. (2009). Graspables: Grasp-Recognition as a User Interface. *In Proc. ACM CHI '09.* p. 917-926.

[23] TrackPoint Mouse. (2004). IBM Research. http://www.almaden.ibm. com/cs/user/tp/tpmouse.html. (Sept. 2009).

[24] Villar, N., Izadi, S., Rosenfeld, D., Benko, H., Helmes, J., Westhues, J., Hodges, S., Butler, A., Ofek, E., Cao, X., and Chen, B. (2009). Mouse 2.0: Multi-touch Meets the Mouse. *In ACM UIST'09.* 33–42.

[25] Venolia, G. (1993). Facile 3D Direct Manipulation. *In Proc. ACM CHI '93.* 31–36.