Mouse 2.0: Multi-touch Meets the Mouse

Nicolas Villar[1], *Shahram Izadi*[1], *Dan Rosenfeld*[3], *Hrvoje Benko*[2], *John Helmes*[1],
Jonathan Westhues[3], *Steve Hodges*[1], *Eyal Ofek*[3], *Alex Butler*[1], *Xiang Cao*[1], *Billy Chen*[3]

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

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

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

{nvillar, shahrami, danr, benko, jhelmes, jonawest, shodges, eyalofek, dab, xiangc, bilchen}@microsoft.com

Figure 1. Our multi-touch mice explore different touch sensing techniques, form-factors and interactive affordances.

ABSTRACT

In this paper we present novel input devices that combine the standard capabilities of a computer mouse with multi-touch sensing. Our goal is to enrich traditional pointer-based desktop interactions with touch and gestures. To chart the design space, we present five different multi-touch mouse implementations. Each explores a different touch sensing strategy, which leads to differing form-factors and hence interactive possibilities. In addition to the detailed description of hardware and software implementations of our prototypes, we discuss the relative strengths, limitations and affordances of these novel input devices as informed by the results of a preliminary user study.

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

General terms: Design, Human Factors, Algorithms

Keywords: Multi-touch, mouse, surface computing, desktop computing, novel hardware, input device.

INTRODUCTION

Humans are naturally dexterous and use their fingers and thumbs to perform a variety of complex interactions to a high precision. The traditional computer mouse design, however, makes little use of this dexterity, reducing our hands to a single cursor on the screen. Our fingers are often relegated to performing relatively simple actions such as clicking the mouse buttons or rolling the mouse wheel.

With the emergence of multi-touch, we now have the opportunity to manipulate digital content with increased dexterity. But whilst multi-touch has been incorporated into many different form-factors – from tabletop to mobile phone – it has yet to find a place on our desktops. This may seem surprising, particularly given that for many computing tasks the desktop setting still dominates.

In this paper, we explore the possibilities for bringing the benefits of multi-touch interaction to a traditional desktop setting, comprising of a computer, vertical display, keyboard and mouse. Given the prevalence of the mouse on the desktop, we feel there is an opportunity to extend this input device with multi-touch capabilities. We refer to these novel input devices as *multi-touch (MT) mice*. In addition to serving as devices for common pointer-based interactions, MT mice conceptually allow the user to reach into the GUI – enabling them to manipulate the graphical environment with their fingers, and to execute commands via hand-gestures without the need to physically touch the display.

The main contribution of this paper is a technical exploration of the design space for MT mice through five different hardware prototypes. After reviewing related work in this area, we present a detailed description of the technical implementation of each of our MT mice prototypes, focusing first on the hardware and then on the low-level sensing and processing. We go on to describe how MT mice are used to enhance the desktop user experience, and conclude by discussing the possibilities afforded by the different mice implementations and their relative merits, as informed by the results of a pilot user study.

RELATED WORK

The basic computer mouse design has remained essentially unchanged for over 40 years following its first public demonstration by Doug Englebart et al. [9]. Since then, repeated efforts have been made to augment the basic mouse

UIST'09, October 4–7, 2009, Victoria, British Columbia, Canada.

functionality with additional capabilities. Arguably, the most successful addition has been the scroll wheel [30] which was originally added to support 3D interactions.

One of the primary areas of research in this space has focused on extending the number of degrees-of-freedom (DOF) that the mouse can sense and thereby control. MacKenzie et al. [19] and Fallman et al. [10] describe prototype devices that contain hardware from two mice rigidly linked into a single chassis to enable rotation sensing and thereby provide 3DOF input. Rockin'Mouse [2] augments a mouse with tilt sensors to enable 4DOF input. The bottom of the device is rounded to facilitate this rocking motion, which is used to control the two extra DOFs for manipulation of 3D environments. VideoMouse [14] is a mouse augmented with a camera on its underside and employs a mouse pad printed with a special 2D grid pattern. It uses computer vision to detect changes in the grid pattern to support full 6DOF input, including tilt, rotation and limited height sensing. Manipulating the mouse in mid-air is also possible with mice that include accelerometers and gyroscopes (e.g., [1][12]).

Cechanowicz et al. [7] investigated the use of uni- and dual-pressure augmented mice, where one or more pressure sensors mounted on the mouse simultaneously control cursor position as well as multiple levels of discrete selection for common desktop applications. Kim et al. [17] investigated the concept of an inflatable mouse which could also be used for pressure sensitive input.

PadMouse [3] adds a touchpad on top of the mouse. This single-touch sensing prototype demonstrates the benefits of such a configuration in precise pointing tasks. Similar benefits can be achieved by substituting the absolute position-sensing touchpad for a relative-position sensing mini joystick (e.g. TrackPoint Mouse [29]) or a miniature trackball (e.g. MightyMouse [23]). In contrast to our work, these approaches only support single fingertip input.

While not allowing for multi-touch interactions on a single mouse device, Latulipe et al. [18] have investigated symmetric bimanual input performed with two mice in conjunction with a desktop display, finding this superior to the asymmetric case or using a single mouse. Absolute sensing of the mouse location on the surface has been explored in the FieldMouse project [28].

Our work also draws inspiration from numerous interactive surface products and prototypes which enable multi-touch interactions through either embedded electronics (e.g., [8][26]) or camera-based methods (e.g., [21][13][22]). Forlines et al. [11] evaluated the benefits of direct touch vs. standard mouse input for interactive tabletops and found overall preference for direct touch, but noted that mouse input might be more appropriate for standard applications requiring precise single-point input.

There are of course other ways to bring multi-touch interactions to the desktop, rather than augmenting the mouse. For example, it is possible to augment the vertical display with direct input capabilities. There have been several attempts to mitigate the resulting precision and occlusion problems [27], for example using bimanual multi-touch interactions [5] or placing the contacts behind the screen [33]. However, this is still not the most ergonomic configuration for desktop use – user's hands and arms will quickly fatigue and users have to explicitly switch between using the touchscreen and the mouse for input.

The benefits of multi-touch interactions can also be achieved with multi-touch sensitive pads that are not coupled with the display (e.g., [15] or the touchpad in Apple laptops). Malik et al. [20] have also explored how a camera-based multi-touchpad can be used for indirect input to large displays. Moscovich et al. have developed a number of multi-finger interaction techniques and graphical cursors for multi-touch pads [25]. However, as surveys have revealed, most users prefer mice to touchpads, especially for precise selection tasks [16].

Given the limitations of touch screens and pads for bringing multi-touch to the desktop, we explore a different approach for enabling such capabilities, which takes the mouse as the starting point. The mouse has gone through several decades of iterative refinement; it offers high resolution pointing, is ergonomically designed to be held in a single hand and requires little effort to use. It is a well-established device for the desktop and we feel that there are opportunities for complementing the capabilities of regular mice with the compelling new interactions afforded by multi-touch systems.

HARDWARE DESIGN PROTOTYPES

MT mice are conceptually very simple. The main novel contribution of our work is in investigating how we can realize such devices in practice. This section provides a technical exploration of the design space for MT mice, with the aims of broadening our understanding of how to build such novel input devices. We present five MT mouse hardware devices; each presents a different implementation and sensing strategy that leads to varying device affordances and form-factors, and hence very unique interaction experiences.

One of our main goals when realizing these mice is to support multi-touch gestures *alongside* regular mousing operations. MT mice should therefore still allow the user to easily *grasp* and *release* the device, *move* it with their wrist or forearm, *clutch* it for repositioning, and perform standard cursor interactions such as *clicking*, *dragging* and *selection* without compromising precision.

We describe each prototype in turn, explaining our motivations and rationales behind each design, and outlining the hardware design and key implementation details.

FTIR Mouse

Our first MT mouse design is based on a common technique for enabling multi-touch input on interactive surfaces: frustrated total internal reflection (FTIR) [13]. With this approach a sheet of acrylic is edge-lit with infrared (IR) light. When a finger is pressed up against the acrylic, it causes IR light to be scattered away from the finger; this can be detected using an IR camera which is imaging the surface. Although this technique has been used to provide multi-touch for a variety of systems, our approach applies FTIR to the surface of an indirect input device, augmented with a regular mouse sensor. Our FTIR Mouse is shown in Figure 2.

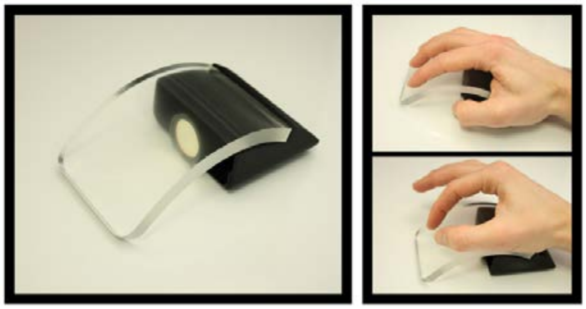

Figure 2. FTIR Mouse applies the principle of frustrated total internal reflection to illuminate a user's fingers, and uses a camera to track multiple points of touch on its curved translucent surface.

In order to adapt this technique into a form-factor suitable for a mouse, we molded a piece of acrylic into a smooth arc shape. The acrylic arc is mounted into a custom base containing a row of IR LEDs, in such a way that the edge of the arc is pressed flush against the LEDs. The base also contains a standard optical mouse sensor to track its displacement across the surface, as well as a small PGR FireFly MV camera equipped with a wide-angle lens, mounted so that it captures the underside of the arc in its entirety.

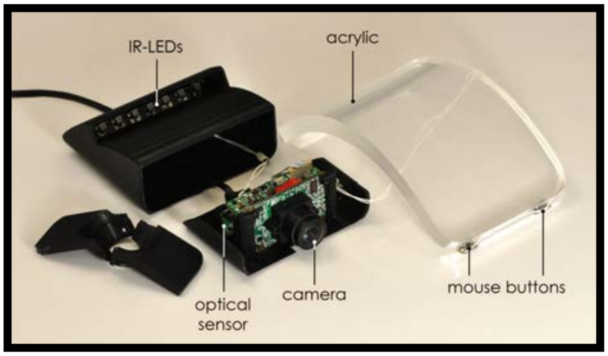

Figure 3. The main components of FTIR mouse.

Small buttons mounted under the forward edge of the arc are linked to the internal mouse circuitry, allowing the user to perform mouse-clicks by depressing the front of the device with their fingers. Figure 3 shows the main components of the FTIR mouse.

Figure 4 shows an example image captured by the camera when a user touches the front surface of the arc with three fingers. These touching fingers are clearly illuminated as they touch the surface because they cause some of the IR light to scatter to the camera. These images are processed using a vision pipeline described later, to derive the position of touching fingers.

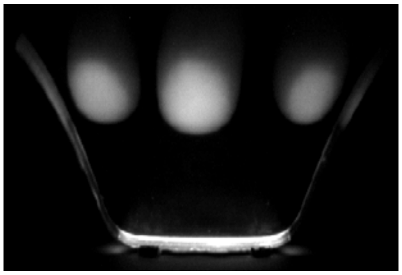

Figure 4. The IR camera with wide angle lens captures the front of the FTIR mouse. In this example, the three fingers touching are illuminated.

We found the FTIR approach to be a suitable technique for prototyping MT sensing on the surface of the mouse. FTIR inherently gives a binary indication of when the user is touching the surface which makes the interaction robust in operation. Also, from an industrial design perspective, the use of the clear acrylic affords some interesting aesthetic possibilities. However, the FTIR technique does have limitations as a means of sensing multi-touch, and places some restrictions on the physical design of the device (which may be at odds with ergonomic requirements). For example, sensing is limited to the area at the front of the device (in the camera's field of view), meaning that only the user's outstretched fingertips can be sensed. The use of an IR-sensitive camera as a sensor makes the device susceptible to sunlight and other external sources of IR light – a well-known problem for camera-based interactive surfaces. Furthermore, the shape and curvature of the transparent acrylic section cannot be chosen arbitrarily, as steep curves or a convex outline would break the total internal reflection. In order to address some of these limitations, our next prototype explores an alternative hardware implementation: the use of *diffuse* IR illumination to track a user's hands on a surface, coupled with additional optics which extend the field of view of the camera.

Orb Mouse

Orb Mouse is shown in Figure 5. It facilitates multi-touch sensing on its hemispherical surface by incorporating an IR-sensitive camera and internal source of IR illumination. Unlike FTIR Mouse, the illumination is not totally internally reflected through the shell of the device; rather, it radiates outwards from the centre of the device, and is reflected back into the camera by objects (such as the user's hands) that come into close proximity to the hemispherical surface of the mouse.

The basic principle of operation is similar to the certain interactive surface technologies that use diffused IR illumination (e.g., [21]). Figure 6 illustrates the internal construction of our prototype. We again use a PGR FireFly MV camera with an IR-pass filter, together with four wide-angle IR LEDs as the illumination source. Instead of pointing directly at the surface, the camera is aimed towards an internally mounted hemispherical mirror. This gives the camera a very wide angle view of most of the mouse surface. Folding the optics in this way also has the benefit of maintaining a relatively low-profile form-factor which is critical if the device is to be used in a mouse-like manner.

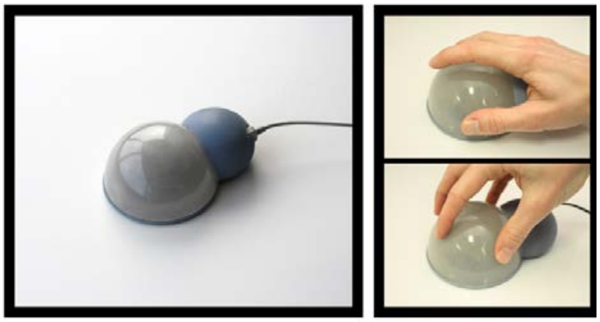

Figure 5. Orb Mouse is equipped with an internal camera and a source of diffuse IR illumination, allowing it to track the user's hand on its hemispherical surface.

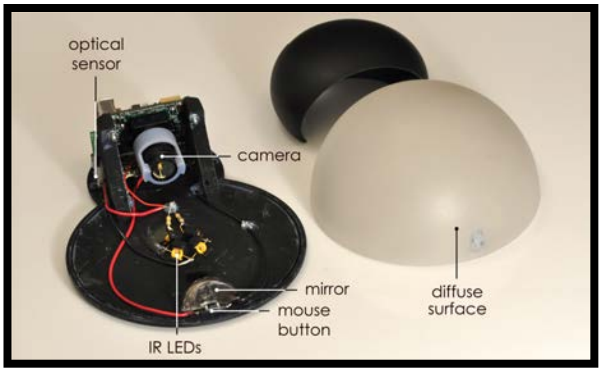

Figure 6. The main components of Orb Mouse.

In addition to the MT-sensing components, standard optical mouse circuitry is integrated to the base of the device to track its displacement across the surface and a microswitch is placed underneath the mirror to allow the entire hemisphere to be "clicked".

As shown in Figure 7 left, the camera image of the reflector is heavily distorted; much more so than with FTIR mouse. We undistort this image using a technique similar to that described in [4]. This corrected image is further normalized (to account for non-uniform illumination across the surface), binarized, and finally a connected component analysis is used to locate the centre of any touching contacts. This pipeline is highlighted in Figure 7. Note that FTIR mouse uses a similar vision pipeline, although the initial correction of the image is based on different optical geometry.

The hemispherical shape of Orb Mouse is intended to be relatively easy to grip and the constant curvature ensures that the user's fingers experience a smooth gradient while moving from side to side and front to back. In addition, Orb Mouse's actively-sensed interaction area is substantially larger than that of FTIR Mouse, encompassing both the top and sides of the devices, thereby allowing all fingers and even the whole hand to be engaged in interactions.

As with the FTIR Mouse design, the Orb Mouse is sensitive to IR light from external sources. Although the use of diffuse illumination coupled with folded optics affords greater flexibility in form-factor, it is also much more noisy and susceptible to interference; the reflected-IR images of the user's touch-points, as captured by the camera, are considerably lower contrast than those possible with an FTIR design. To overcome some of these issues, we have also explored alternatives to camera-based sensors, described in the next section.

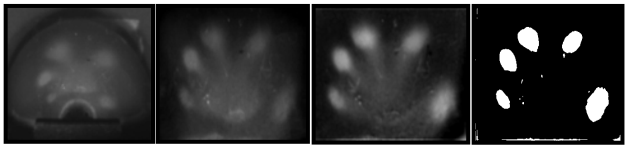

Figure 7. The vision pipeline for Orb Mouse. From left to right: the image is captured from the mirror, undistorted, normalized and binarized before the position of individual contacts are calculated.

Cap Mouse

The Cap Mouse (short for *capacitive* mouse) prototype tracks the position of multiple fingers on its surface through capacitive touch sensing as shown in Figures 8 and 9. This prototype uses a flexible matrix of capacitive-sensing electrodes to track the location of the user's contacts.

Figure 8. Cap Mouse employs a matrix of capacitive touch-sensing electrodes to track the position of the user's fingertips over its surface.

In contrast to previous designs which use capacitive sensing for detecting clicks only [23], 1D scrolling [1], or the single finger position [3], the Cap Mouse design is novel in that the device includes a true multi-touch sensor and thus is able to simultaneously track the locations of all of the user's fingers on the surface of the mouse. In addition to capacitive multi-touch sensing, the base of the mouse contains a regular mouse sensor and single mouse button which can be clicked by pressing down towards the front of the device.

Figure 9 illustrates the internal components of our prototype. An X-Y grid of sensing elements is printed on a flexible plastic substrate using conductive ink. This sensor is wrapped around the front portion of the mouse's surface and covered with a thin plastic shell to prevent direct electrical contact between fingers and the sensor elements. When a user's finger is placed on the shell it affects the mutual capacitance between the sensing elements of nearby rows and columns. This can be detected and pinpointed using a microcontroller which sequentially scans the various combinations of rows and columns.

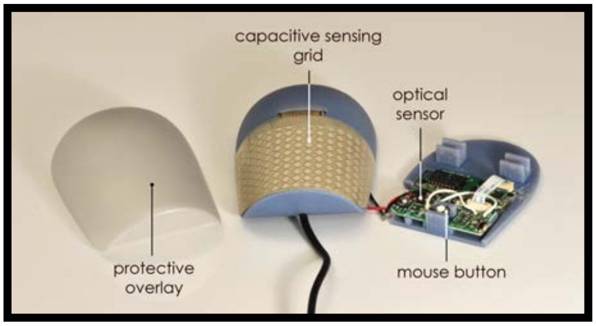

Figure 9. The main components of Cap Mouse.

Capacitive sensor elements are placed with center-to-center spacing of around 5 mm but the effective resolution of the sensor is actually much finer than this – a finger will generally cover multiple sensor elements so it is possible to interpolate position from multiple capacitance readings. The raw capacitive sensor values are converted into a 20x10 grayscale image, which is interpolated up by a factor of 10 along each axis. This image is then fed through part of the same vision pipeline as FTIR and Orb Mice, from binarization onwards, to extract the location of touching fingers.

The capacitive approach is an appealing means of constructing a surface touch sensor. Unlike our optical mice, it is immune to ambient illumination. The sensor provides much less data than the cameras included in other designs—thus lowering bandwidth and processing requirements—while still allowing good positional precision through interpolation. Cap mouse is also physically more compact because the design constraint imposed by the optical path required in our vision-based prototypes is eliminated. The compactness of the sensor enabled us to design a mouse with a relatively conventional form and scale, and thus investigate the pros and cons of performing multi-touch gestures on an otherwise normal mouse. It also has relatively low power consumption. However, the effective resolution of the capacitive sensor is considerably lower than with a camera-based approach.

Side Mouse

The previous three prototypes augmented the surface of a mouse with multi-touch sensing capabilities, however other designs are also possible. The Side Mouse device senses the user's fingers as they touch the table surface *instead of* the mouse. This design is inspired by [6] which explored proximity based sensing around the periphery of mobile devices. Side Mouse is designed to rest under the user's palm, allowing fingers to touch the table surface directly in front of the device as shown in Figure 10.

The key components of the device are highlighted in Figure 11. The base is equipped with a forward-sensing camera, mounted behind an IR-pass filter. Underneath the camera, and suspended a few millimeters above the surface, sits a line-generating IR-laser illuminator which casts a sheet of IR light that fans outwards from the front of the device. An ultra-wide angle lens allows the camera to image the area covered by the illuminator. Fingers and other objects placed in this area reflect IR light back to the camera, allowing it to sense their positions as shown in Figure 12. These images are once again processed using the same vision pipeline presented earlier.

Figure 10. Side Mouse rests under the palm of hand, allowing fingers to touch the table surface directly in front of the device. These are sensed using an internal camera and IR laser.

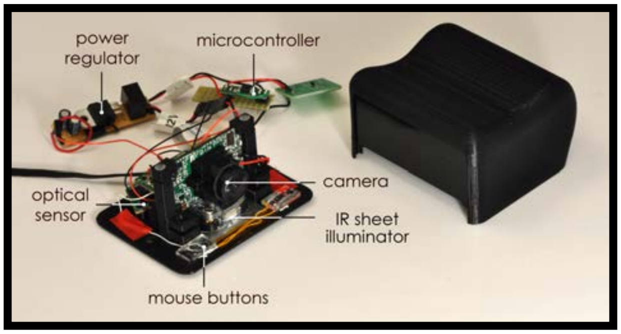

Figure 11. The main components of Side Mouse.

In addition, the base of the device is equipped with an optical mouse sensor allowing it to carry out regular pointing tasks. Since the user's fingers rest directly on the surface, performing a mouse-click with this device is achieved by pressing it down with the palm of the hand. This action is detected by a pair of buttons mounted on the underside of the case, which require an actuation force high enough that they are not triggered accidentally when simply resting a palm on the device, yet not so high to require an undue amount of force to perform a mouse click.

Figure 12. The Side Mouse camera image. Two fingers are hovering above the surface and are not illuminated (left). Once touching the surface, the IR beam is broken by the fingers and these become brightly illuminated (right).

The key interaction possibility that we explore with Side Mouse is the ability to create a multi-touch area that is not restricted to the physical surface of the device. This approach allows the input space to be larger than the physical bounds of the device. This wide sensing scope does however have practical implications in detecting stray objects (not just fingers) like the keyboard and other items on the desk. As well as interactions whilst the device is ‘gripped’, the main body of the mouse can also be ‘parked’ – that is, moved to a location and subsequently released. This defines an ad-hoc interactive region on the surface of the desk where both hands can be used for bimanual multi-touch interactions.

Arty Mouse

Side Mouse opens up the interaction space around the mouse. However, like our other camera-based mice it has issues regarding susceptibility to lighting and higher power consumption. Our final prototype, which we call Arty Mouse (short for *articulated* mouse*)*, takes the notion of Side Mouse one step further.

Figure 13. Arty Mouse is equipped with three high-resolution optical mouse sensors: one in the base, which rests under the user’s palm, and two under the articulated extensions that follow the movements of the index finger and thumb.

In our Arty Mouse design (shown in Figure 13), the palm of the hand rests on the base of the device; from this base extend two articulated ‘arms’ that can be freely and independently moved on the table by the thumb and index finger. The design makes use of three separate optical mouse sensors – one under the base and one underneath each articulated arm – to individually track the displacement of each of these parts across the surface. The design is tailored towards use with the thumb and index finger, although other finger arrangements are also possible.

The base of the Arty Mouse houses the circuitry from three separate Bluetooth optical mice as shown in Figure 14, making this our only wireless MT mouse currently. For our prototype we chose to use components extracted from MoGo mice [24], due to their extremely thin form factor. The optical sensors on these devices were decoupled from their original circuitry and re-soldered to a small (2cm diameter) PCBs of our own design, which includes some passive components necessary for correct operation. One of the sensors is placed on the underside of the base, and the other two at the end of the articulated arms.

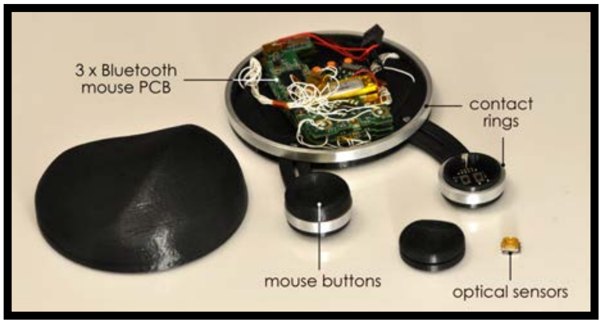

Figure 14. The main components of Arty Mouse.

The arms are attached to the base using a grooved pivot mechanism, allowing them to be moved with 2DOF while still retaining some desirable physical constraints that help maintain mechanical stability while moving and clutching the device. In addition to the mouse sensor, each arm is equipped with a button that can be depressed with the fingertips. Conductive metal rings surround the base of the device as well as the extremities of the arms. These act as contact sensors, detecting whenever two of these parts (the base and index part, the base and thumb part, or the thumb and index part) are touching.

It is important to note that the mice sensors are *relative position* devices, but for some applications *absolute position* of each part with respect to each other may be desired. This could be achieved using additional sensors on the articulated arms – such as rotary and linear encoders or potentiometers – to estimate their angle and distance from the base. However, for the sake of simplicity in an already complex mechanical design we opted instead for a dead-reckoning software technique, where the relative movement vectors of the arms are summed to estimate their current position with respect to the base. With this technique it is important to establish a ground-truth (a known position from which to start measuring changes in position), and for this we bring into play the metallic contact rings: when the base, index or thumb touch each other, which happens regularly and naturally during interaction with the device, this gives an indication of their absolute position along one axis.

One key advantage of this particular design over other sensing techniques explored in this paper is the fact that it allows a high-resolution optical mouse sensor to be placed underneath two of the user’s fingers. This technique provides extremely high sensing fidelity compared with capacitive or camera-based sensing techniques described earlier, and can be leveraged to support subtle and fine-grained multi-touch gestures.

COMBINING MOUSE INPUT WITH MULTI-TOUCH

Enriching the Cursor with the Multi-touch Cloud

In order to accommodate multi-touch interaction - while still supporting traditional mousing - we have developed an augmented version of the standard GUI mouse cursor, called the Multi-touch (MT) Cloud. This is just one simple method for combining the absolute data derived from the touch sensor with the pointer-based input of the mouse (other techniques such as those described in [25] are also feasible).

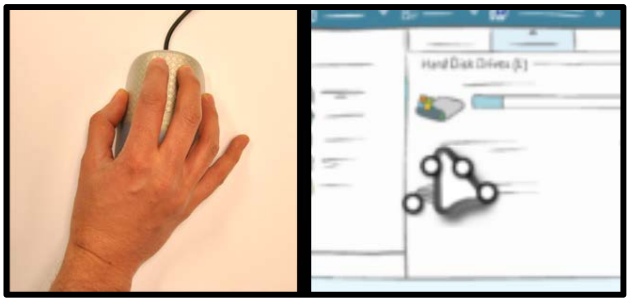

Figure 15. Multi-touch points on the mouse are mapped to a small region around the cursor, and this data is piped directly to the window in focus. These are visualized as small dots around the cursor. Here the user is touching the device with thumb and three fingers, indicated as four dots on the cursor.

As with a regular cursor, the on-screen position of MT Cloud cursor is driven by the displacement of the mouse on the surface. This allows the user to carry out regular pointing, selection and dragging operations by moving the mouse and by pressing the physical clicker.

Regular multi-touch sensors typically map touch to the absolute bounds of the screen. So, for example, when a user touches the bottom right of the sensor the touch event occurs on the bottom right of the screen. Our approach differs, in that when touch points are sensed, they are mapped to a small area around the current position of the cursor, as shown in Figure 15.

For example, if the user places a finger on the center of the Cap Mouse touch sensor, a touch event will be generated wherever the centre of the cursor is currently. If the user then removes the finger and moves the mouse physically, the cursor is displaced. Placing the finger on the touch sensor again at the center will cause another touch event to be generated, but this time at the center of the new cursor position. If the user was to touch the far middle left and right of the Cap Mouse touch sensor simultaneously, touch events would be generated to the left and right of the current cursor position and so forth.

This approach ensures that regular mousing and multi-touch can seamlessly coexist. Multi-touch data will only be sent to the window directly under the cursor, thus maintaining the notion of input focus that is familiar to users of desktop interfaces. In our current implementation, these touch events are injected into the message queue of the window with focus, allowing applications and widgets to process these events alongside regular mouse and window events. The raw absolute touch points are also encapsulated in these events, which may be required for certain MT gestures such as Rotate-Scale-Translate (RST).

Finally, to provide feedback to the user, corresponding dots indicating the position of the touch events are rendered on and around the cursor.

Multi-touch Applications on the Desktop

One initial goal for our MT mice implementations was to use them to interact with existing multi-touch applications on the desktop. In so doing we have built a generic bridge between the Microsoft Surface SDK [22] and our MT devices. This allows each of our devices to inject their MT data into regular Surface applications using an implementation of the MT cloud technique. A variety of surface applications, while not specifically designed for the desktop, are able to work in these settings ‘out of the box’. We support standard multi-touch gestures for RST, flicking and so forth, without compromising standard mousing operations such as pointing, selecting and dragging. By using the $1 Gesture Recognizer technique [32], we have also implemented a way to perform single touch stroke-based gestures, in a way that we feel is more comfortably and accurate than gesturing by moving the entire mouse.

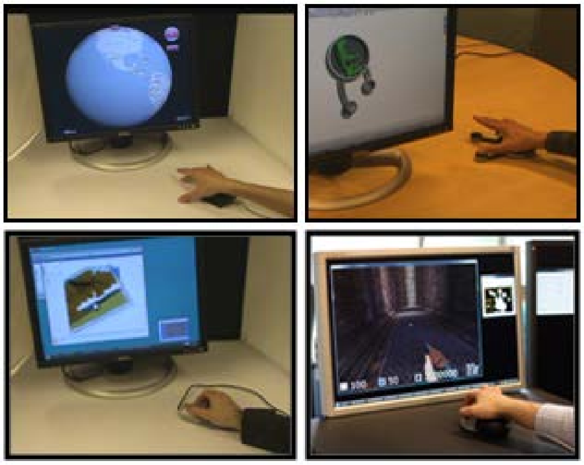

Figure 16. Applications running with our MT mice. Clockwise from top-left: manipulating Virtual Earth, 3D modeling in SolidWorks, controlling a first person shooter game and photo browsing using a desktop mockup.

We have only begun to explore the use of these devices for specific applications, which will be the focus for our future work. One interesting possibility is the ability to map the additional DOFs that are opened up by multi-touch to more intuitive 3D manipulations. We have demonstrated the use of Arty to allow both cursor control and 3D camera manipulation in the SolidWorks CAD application using the additional articulated sensors, each offering an additional 2DOF. We have also explored mapping the rich MT data sensed from Orb Mouse to a first person shooter game, allowing simultaneous control of the virtual camera in 6DOF and other controls that would typically be associated with keyboard shortcuts, such as changing weapons. These are shown in Figure 16.

PILOT STUDY

In order to better understand the affordances of each device, and get an initial sense of their relative strengths and limitations, we conducted a pilot user study. We asked a group of 6 users to repeat a structured task, namely using each of our devices in turn to rotate, scale, and translate a randomly placed image to approximately match a target frame. The MT cloud technique was used throughout the experiment. At this stage of the work, given the broad questions surrounding the ergonomics and capabilities of our devices, we chose to focus on the qualitative aspects of the user experience, as opposed to generating quantitative results. We discuss the observations from these early-stage studies and some broader questions about the design of MT mice in the following sections.

Each user tried each of the 5 devices in sequence. After the user finished with each one, we conducted a brief interview. The user was also asked to rate the device in terms of general feeling, physical comfort, interaction intuitiveness and ease of use in comparison to other devices that had been used. The users were encouraged to think aloud during the whole process. We directly observed and video-recorded the users' behaviors when using the devices. Six volunteers, 5 right-handed and one left-handed, participated in the evaluation. These included both people with little previous experience with multi-touch input and those who have used it extensively.

Observations

All participants were able to use the 5 devices, and managed to complete the task in relatively short times. The MT cloud model seemed understandable. For example, users intuitively moved the mouse over the target before performing MT actions, if the cursor moved slightly off the target and their MT actions were ignored, they quickly noticed and moved the cursor back on target. This indicated that our users had little problem adapting to this hybrid model for input, which brings notions of cursor input and multi-touch together.

Arty was received the most positively by the users in terms of general feeling, physical comfort and ease of use. In many ways, this is understandable given that the two articulated points on Arty sit comfortably under the thumb and index finger, making it extremely easy to carry out pinch gestures. We observed users very naturally carrying out these actions. The high precision of the sensors underneath each articulated point made it a very accurate device for the task. Users were able to simultaneously carry out RST gestures, which was not the case for any of the other devices, and this coupled with the high accuracy led to rapid and fine control. Responses in the questionnaire also highlighted that the articulated points added to the users comfort. However, Arty only supports two points of multi-touch input, which although sufficient for our experiments, limit the multi-touch gestures possible.

Interestingly, Orb Mouse was also a very popular choice. Users found the device's form-factor and affordances led naturally to RST gestures. However, rather than using a pinch gesture to perform scale and rotate, all users found that rotation was more comfortable being performed using all five fingers to grip the device and rotating these left and right in a lateral motion. For scaling up, users placed all five fingers on the top of the device and moved these down towards the base of the device (and vice-versa, when scaling down). These interactions make full use of the device's 3D shape and map well to its hemispherical form. Unlike most of the other devices, we saw the most apparent learning curve in using this device, as users 'got to grips' with this new technique for rotate and scale.

Users found many aspects of Side Mouse compelling, in particular leveraging a larger touch surface for MT input. They however struggled with the form-factor of the current implementation. Given the diversity of hand sizes, the device was too tall for smaller hands to touch the surface whilst simultaneously resting the device under their wrist. Users with larger hands, often found their fingers were 'out of range' of the sensor whilst their palm was resting on the device. This led to fairly bimodal use of the device – it was typically gripped one way for mousing and then the grip was changed to perform multi-touch input. The other problem was that users felt it was uncomfortable to activate the clicker in the base of the device whilst simultaneously moving the device. This suggests a 'virtual' clicker based on the multi-touch sensor data may prove more effective. None of these limitations are insurmountable, but they do highlight how critical form-factor is in realizing such MT devices.

Cap Mouse was also deemed as appealing given the mouse-like form-factor. Users found it a familiar device to interact with. It was also the smallest of our mice making it easier for users to grip and clutch.

DISCUSSION

Being Mouse-Like

One clear aspect to emerge from our study is the importance of ergonomics and form-factor. For some devices we spent a considerable time on the form of the device (e.g. Arty and Cap Mouse), while for others the form is merely a byproduct of the technology used (e.g. Side Mouse); this was reflected in users' experiences with the device. While there is clearly more work to be done in regards to form-factor, one of the interesting findings from our work is how receptive our users were to devices that move away from the established mouse form-factor. Initially we had hypothesized that users would only be receptive to mice that closely replicate a traditional mouse shape, but in fact they were open to more exotic designs.

Interestingly the 'mouse-like' design of Cap Mouse led users to have certain biases, based on their existing experiences using a standard computer mouse. For example, initially and without practice, when users were asked to rotate an onscreen object using Cap Mouse they would often use a single finger as if they were using a virtual mouse wheel. This of course failed to rotate the object. It took them several attempts to learn that such an interaction requires a pinch using thumb and forefinger.

Our observations of Cap Mouse also show benefits when leveraging a more traditional mouse form-factor, in particular making regular mousing comfortable. However, we also saw value in moving away from a traditional mouse form-factor as evidenced with Orb Mouse. Here, based on interview feedback, we get the sense that users thought differently about the capabilities of the device simply because it looked and felt qualitatively different to a mouse, which led to more experimentation with MT gestures. There is a tradeoff here however, as users hands began to fatigue over time when using this device. Here FTIR mouse seemed to strike the right balance between the ergonomics for mousing and touch.

To Click or Not to Click

All our devices had physical clickers embedded in them to support standard selection tasks. We had originally considered using the clicker to explicitly 'turn on' MT gestures, but after initial testing we felt this would be too limiting for the user. However, clicking to enable MT seemed intuitive to our users – they commented that activating MT while not clicking would be '*strange*'. However, we also observed problems leading from the need to physically press and gesture at the same time. It becomes very difficult to move fingers that are also pressing the device down.. Typically this leads to one finger pressing down whilst using the others to carry out the MT gesture. This clearly is a limitation in terms of supporting more complex multi-fingered gestures. Further the friction caused by pressing down on the device, also makes it difficult to move the mouse whilst using touch, leading users to switch 'modes' between multi-touch and mousing.

Expose the Sensing

Another important design challenge to emerge from the study was the need to physically expose the MT sensing area of each device. This was the most apparent for Side, followed by FTIR Mouse, where users struggled to know if fingers were within the sensing area. One option specific to Side Mouse would be to use a projected laser pattern to demarcate the sensing area on the desktop. However, even for devices such as Cap Mouse where the demarcation was clearly marked, oftentimes users did not realize exactly when their fingers were being registered as touch points. This is perhaps because they rarely looked down at the mouse during desktop interactions, and so were not completely clear about where the sensing area started and ended. A bezel, such as those used on regular touch pads, could have helped by giving the user more tactile feedback. We have also begun explore how this type of feedback could be provided in the UI, by way of more expressive cursor designs, such as the MT cloud described earlier. Finally, in terms of physical design of the device, it seems important to provide inactive areas where the user can comfortably rest their fingers while clutching the device without accidentally triggering the MT input.

All About Constraints

One of the main comments from users was that some of the devices provided 'too much freedom'. We had anticipated that this additional freedom would lead to more open interactions, but conversely users sometimes felt less comfortable experimenting and interacting because they simply could not predict how they should interact with the device. A clear exception here was Arty, whose physical form afforded particular places to rest fingers and palm, and its physical constraints clearly indicated the range of gestures that were possible. Rather than limiting our users, they realized they were holding and interacting with the device in the manner it was designed for, and seemed comfortable to experiment within that design. Obviously users can be trained to use more open-ended devices, but this finding suggests that molding the shape of the device to suggest the ways that it might be held and interacted with might reduce its initial learning curve.

Don't Mode Me In

It also became apparent from the user study that some of our devices are inherently bi-modal in their support of MT versus regular mousing. This modal nature was particularly clear for FTIR and Side mice and led to occasional frustrations. Users would often 'posture switch', gripping the device in one way to perform MT gestures and another to perform pointing tasks. This was mainly due to the fact that the thumb and forefinger were primarily used to carry out the gestures, and that the form-factors of these devices required the thumb to be repositioned on the side of the device in order to grip it and move it.

One of the interesting challenges of placing a multi-touch sensor on the surface of the mouse is that the device needs to be able to be gripped for regular mousing. This can lead to accidental triggering of the MT sensor just by virtue of the user holding the device. We solved this issue currently by only triggering touch when physically clicking the device, but have found that this leads to moded styles of interaction, as well as other limitations discussed earlier. We feel this is a key challenge to address in the future.

CONCLUSION

This paper has explored a number of ways of introducing multi-touch capabilities to the standard computer mouse. The goal is to make multi-touch interaction more widely available and applicable to the desktop environment. We have begun to chart the design space for novel types of input devices that combine regular mousing capabilities with MT gestures. The core contribution of our work is a technical one – we have established the feasibility of building multi-touch mice, and have documented a variety of approaches for doing so. However, the exercise of building these prototypes has been valuable to us beyond the resulting set of devices. Through the process of design and development, we have come to experience first-hand the tension between technical challenges and ergonomic requirements that lie at the heart of making MT mice practical and desirable.

More concretely, our contributions include: a practical comparison of five different techniques for enabling multi-touch on the desktop, which include three distinct camera-imaging approaches, the use of capacitive sensors to track multiple fingers on a curved surface, and an approach for tracking finger movements using multiple optical mouse sensors; and, our reflections on the general issues regarding MT mice – informed both by the insights gained from our design and development efforts, as well as through the initial user feedback from our preliminary study. In future work, we plan to refine our prototypes – both ergonomically, and in terms of their sensing capabilities – to deeper explore the interaction techniques that are specific to these new class of input devices.

REFERENCES

1. Air Mouse. Logitech. http://tinyurl.com/3qtor8. (last accessed March 2009).
2. Balakrishnan, R., Baudel, T., Kurtenbach, G., and Fitzmaurice, G. (1997). The Rockin'Mouse: Integral 3D Manipulation on a Plane. *Proc. of ACM CHI '97.* p. 311–318.
3. Balakrishnan, R. and Patel, P. (1998). The PadMouse: Facilitating Selection and Spatial Positioning for the Non-Dominant Hand. *Proc. of ACM CHI '98.* p. 9–16.
4. Benko, H., Wilson, A., and Balakrishnan, R. (2008). Sphere: Multi-Touch Interactions on a Spherical Display. *Proc. of ACM UIST '08.* p. 77-86.
5. Benko, H., Wilson, A., and Baudisch, P. (2006) Precise Selection Techniques for Multi-Touch Screens. *Proc. of ACM CHI '06.* p. 1263–1272.
6. Butler, A., Izadi, S., and Hodges, S. (2008). SideSight: Multi-'Touch' Interactions around Small Devices. *Proc. of ACM UIST '08.* p. 201–204.
7. Cechanowicz, J., Irani, P., and Subramanian, S., (2007). Augmenting The Mouse With Pressure Sensitive Input. *Proc. of ACM CHI '07.* p. 1385–1394.
8. Dietz, P. and Leigh, D. (2001). DiamondTouch: A Multi-User Touch Technology. *Proc. of ACM UIST '01.* p. 219–226.
9. Engelbart, D. C. et al. (1968). A Research Center for Augmenting Human Intellect. *(demonstration)* Stanford Research Institute, Menlo Park, CA. http://sloan.stanford.edu/MouseSite/1968Demo.html.
10. Fallman, D. and Yttergren, B. (2007). The Design of a Computer Mouse Providing Three Degrees of Freedom, *HCI Internationa '07l.*
11. Forlines, C., Wigdor, D., Shen, C., and Balakrishnan, R. (2007). Direct-touch vs. mouse input for tabletop displays. *ACM CHI '07.* p. 647–656.
12. Go Pro Air Mouse. Gyration. http://tinyurl.com/l3wmow (last accessed March 2009)
13. Han, J. (2005). Low-Cost Multi-Touch Sensing Through Frustrated Total Internal Reflection. *ACM UIST '05.* p. 115-118.
14. Hinckley, K., Sinclair, M., Hanson, E., Szeliski, R., Conway, M. (1999). The VideoMouse: A Camera-Based Multi-Degree-of-Freedom Input Device. *Proc. of ACM UIST '99.* p. 103–112.
15. iGesture Pad. FingerWorks. http://www.fingerworks.com. 2009.
16. IOGEAR. Laptop Users Prefer Mice over Touchpads, Survey. PC Business Products, 2003.
17. 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. *Proc. of ACM CHI '08.* p. 211–224.
18. Latulipe, C., Kaplan, C.S., and Clarke, C.L.A. (2005). Bimanual and Unimanual Image Alignment: an Evaluation of Mouse-based Techniques. *Proc. of ACM UIST '05.* p. 123–131.
19. MacKenzie, I. S., Soukoreff, R. W., and Pal, C. (1997). A Two-Ball Mouse Affords Three Degrees of Freedom. *ACM CHI Extended Abstracts '97.* p. 303–304.
20. Malik, S., Ranjan, A., and Balakrishnan, R. (2005). Interacting with Large Displays from a Distance with Vision-tracked Multi-finger Gestural Input. *Proc. of ACM UIST '05.* p. 43-52.
21. Matsushita, N., and Rekimoto, J. (1997). HoloWall: Designing a Finger, Hand, Body, and Object Sensitive Wall. *Proc. of ACM UIST '97.* p. 209-210.
22. Microsoft Surface. Microsoft. http://www.microsoft.com/surface..(last accessed March 2009).
23. Mighty Mouse. Apple. http://www.apple.com/mightymouse/. (last accessed March 2009).
24. MoGo Mouse BT. Newton Peripherals, LLC. http://tinyurl.com/yw4lbv. 2009.
25. Moscovich, T. Hughes, J.F. Multi-finger cursor techniques. *Proc. GI '06*, 1-7, 2006.
26. Rekimoto, J. (2002). SmartSkin: an Infrastructure for Freehand Manipulation on Interactive Surfaces. *Proc. of ACM CHI '02*. p. 113-120.
27. Siek, K.A., Rogers, Y. Connelly, K.H. (2005). Fat Finger Worries: How Older and Younger Users Physically Interact with PDAs. *Proc. of INTERACT '05.* p. 267-280.
28. Siio, I., Masui, T., and Fukuchi, K. (1999) Real-World Interaction Using the FieldMouse. *Proc. of ACM UIST '99*. p. 113–119.
29. TrackPoint Mouse. (2004). IBM Research. http://www.almaden.ibm.com/cs/user/tp/tpmouse.html. (last accessed March 2009).
30. Venolia, G. (1993). Facile 3D Direct Manipulation. *Proc. of ACM CHI '93*. p.31-36.
31. Wigdor, D., Forlines, C., Baudisch, P., Barnwell, J., and Shen, C. (2007) Lucid Touch: a See-through Mobile Device. *Proc. of ACM UIST '07*. p. 269-278.
32. Wobbrock, J. O., Wilson, A.D, Li, Y. Gestures without Libraries, Toolkits or Training: A $1 Recognizer for User Interface Prototypes. *Proc of ACM UIST '07.* P.
33. Zimmerman, T., Smith, J., Paradiso, J., Allport, D. and Gershenfeld, N. Applying Electric Field Sensing to Human-Computer Interfaces. *Proc. of ACM CHI '95*. p. 280–287.