TableMouse: A Novel Multiuser Tabletop Pointing Device

Andrew Cunningham, Ben Close, Bruce H. Thomas & Peter Hutterer
Wearable Computer Lab
School of Computer & Information Science
University of South Australia
{andrew.cunningham, ben.close, bruce.thomas}@unisa.edu.au, peter.hutterer@who-t.net

ABSTRACT
This paper introduces the TableMouse, a new cursor manipulation interaction technology for tabletop computing, specifically designed to support multiple users operating on large horizontal displays. The TableMouse is a low-cost absolute positioning device utilising visually-tracked infrared light emitting diodes for button state, 3D position, 1D orientation, and unique identification information. The supporting software infrastructure is designed to support up to 16 TableMouse devices simultaneously, each with an individual system cursor. This paper introduces the device and software infrastructure and presents two applications exposing its functionality. A formal benchmarking was performed against the traditional mouse for its performance and accuracy.

Author Keywords
Interaction, collaboration, collocation, device.

ACM Classification Keywords
H5.2. User Interfaces: Input devices and strategies (e.g., mouse, touchscreen) .

INTRODUCTION
Tabletop computing is an emerging field suited to a number of computing applications (Apted et al., 2006), (Reitmayr et al., 2005), (Patten et al., 2002). These applications are commonly a form of computer supported collaborative work (CSCW) known as single display groupware (SDG), where multiple collocated users interact with a common shared display. Proper interaction technologies are crucial for the usability of these applications and the wide acceptance of tabletop computing.

Unlike desktop displays, where the display is always positioned in a correct orientation to the user, users at a tabletop display may be positioned at any orientation relative to the display. This aspect requires that any interaction device be able to compensate for user orientation. Many relative positioning devices such as the traditional mouse, however, are orientation dependant and require the user to manually specify their orientation.

Tabletop displays require multiple interaction devices operating simultaneously to handle multiple users, as Stewart et al. (1998) discovered that users do not like to share input devices when working collaboratively. Dietz and Leigh (2001) noted the use of multiple mice is problematic for a collaborative environment as it is particularly challenging for users to keep track of different cursors on a large display. Additionally, absolute positioning devices are shown to be preferred to relative positioning devices such as the traditional mouse in providing awareness of intent in collaborative tabletop environments (Inkpen, 2001).

Current input technologies have been utilised with varying success to provide tabletop interaction. Many solutions are restricted to a particular tabletop display technology or custom software, negating other tabletop display technologies and legacy applications. The traditional mouse, ubiquitous in computer environments, suffers from being a relative-positing device with optical tracking technology that cannot operate directly on many tabletop displays (as mechanical mice are not commonly available). Touch-screen technologies are a natural solution for tabletop interaction but can be imprecise and unable to provide consistent unique identification of multiple users. Pen-based pointing devices are natural and precise but lack the number of degrees of freedom of a direct manipulation input device, and once a pen is released from the user's hand the device is unable remain in its current position.

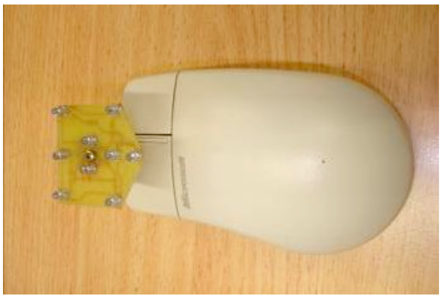

Figure 1. The TableMouse

The contribution of this paper is TableMouse (Figure 1), a low-cost interaction technology suited to the requirements of multiple users working with a shared tabletop display. TableMouse is a pointing device that works as effectively as a traditional mouse with following additional attributes: 1) orientation independence, 2) legacy application support, 3) compatibility with rear/front projected displays and LCD screens, 4) unique identification of multiple devices, 5) absolute positioning, 6) precision, 7) four degrees of freedom (4DOF), and 8) out-of-reach interaction. These aspects have not been provided in a

OZCHI 2009, November 23-27, 2009, Melbourne, Australia.

OZCHI 2009 Proceedings ISBN: x-xxxxx-xxx-x

single solution and in this respect the TableMouse is unique.

This paper begins with a background section covering current research in tabletop interaction technologies. Following, a detailed description of the TableMouse is provided focusing on the functionality, operation, and design of the hardware and software components. We present two applications that begin to explore the functionality of the TableMouse. Finally, we benchmark the TableMouse performance and discuss user feedback and potential applications.

BACKGROUND

As prevalent technologies such as the traditional mouse do not address the requirements of tabletop display interaction, past and current research has looked at applying a variety of solutions. Notable of these are touchscreen input, pointing based devices and tangible interaction devices.

Touchscreen Input Devices

Touchscreen technologies are an obvious choice for interaction devices for tabletop displays. Buxton et al. (1985) note the most critical requirement of touchscreens is that "*the user is not required to point with some manually held device such as a stylus or puck.*" They note the following properties that differentiate touchscreens from other mouse-like devices: 1) the number and type of events transmitted, 2) support for multiple points of interaction, and 3) the ability to act as an assortment of independent virtual devices. A severe limitation they raise is the inability to physically signal to other users while pointing on a touchscreen, while Ryall et al. (2006) note that the low resolution of finger input can make standard GUI widgets unusable when their size is optimised for the more accurate mouse pointer.

Precise software-based selection techniques for touchscreens have been investigated, starting with the *take-off* techniques (Potter et al., 1988). Benko et al. (2006) investigated the following four different forms of dual finger selection: Dual Finger Offset, Dual Finger Stretch, Dual Finger X-Menu, and Dual Finger Slider. They found dual finger selections increased the precision and accuracy in small target selection tasks and, in particular, the increasing of the target size overcame the problem of fingertip occlusion.

Tabletop Pointing Devices

Guimbretière and Winograd (2000) utilised ultrasonic EFI e-Beam pen devices to support the FlowMenu system, which combines command, text and data entry. To support the marking menu functions an extra button had to be added to the pen device. Laser pointers (Olsen and Nielsen, 2001) have been utilized as cursor control devices for large displays. Mouse button presses have been supported through various means: dwell time, strobing the laser, and an external wireless mouse button on the laser.

TractorBeam (Parker et al., 2005) combines the natural interaction of a pen device for graphical objects within arms reach with virtual laser pointing for further objects on tabletop displays. The TractorBeam is a 4DOF device (x, y, z and azimuth) utilising a top-projected display and a Polhemus Fastrak for 6-DOF tracking. Currently, the TractorBeam only supports a single user, and due of the limitations of the tracking technology, the pen device must be tethered that reduces user mobility.

The Sensetable (Patten et al., 2001) electromagnetically tracks the positions and orientations of up to ten wireless objects on a tabletop display surface. The Sensetable employs the Wacom Intuous^{TM} sensing tablet that support 32-bit identification numbers for each device on the tablet. The authors state this form of technology is superior to vision based tracking as it is not susceptible to occlusion or changes in lighting conditions. A novel feature of the Intuous tablets is that tracked objects have state that can be modified by attaching physical dials and modifiers. A limitation, however, is that each tablet only supports two tracked objects natively; the Sensetable multiplexes the tracking by randomly switching the sensing coils in the tracked devices on and off. This multiplexing limits the number and performance of devices on one table; as the number of devices increases, the latency of tracking increases. A second limitation is only two devices may be moved at one time. The Sensetable was extended with the AudioPad (Patten et al., 2002), which allows for detection of rotation of the pucks and can track up to nine pucks simultaneously to an accuracy of 4mm.

Tangible Interaction Devices

Fiducial markers allow 6DOF tracking with unique identification of visually track objects in the physical world (Rekimoto and Ayatsuka, 2000). Rekimoto and Ayatsuka developed their own version of tabletop *phicons* (physical icons), which are tangible objects that provide a means of interaction with a computer. CyberCode's phicons track translation and orientation and are uniquely identified but do not have the notion of mouse buttons. An I/O Bulb (Underkoffler and Ishii, 1998) configuration has been used to track physical objects coded with colour dots on a table surface. This coding scheme allows the position, orientation, and identification of the object to be tracked.

PuckControl

The TableMouse is inspired by Takatsuka et al.'s PuckControl (2006) table-based input device. The PuckControl employs infrared (IR) Light Emitting Diodes (LED) as visual landmarks for a 2D translation and button activation vision based tracking system. IR LEDs were employed as ambient IR levels are commonly quite low in an indoor setting. Four IR LEDs are positioned on the bottom of the puck device arranged in a circle configuration. Devices are operated on top of a back-projected screen, which allows the LED positions to be captured by an infrared camera placed beneath the screen. Two LEDs allow for a cursor position to be calculated. The activation of one of the LEDs signals a button press from that respective mouse button thus providing the advantage of being a wireless/radio-free solution. A major benefit of this scheme is its scalability, as many pucks can be easily supported on the tabletop surface simultaneously without significant latency.

TABLEMOUSE

We have designed the TableMouse as a visually tracked device to meet the eight desired attributes for tabletop interaction previously stated. A computer vision based approach allows the devices to be operated untethered, is not affected by ferrous or electronic interference, and can handle a large number of simultaneously tracked devices. The TableMouse builds upon the concept of Takatsuka et al.'s (2006) visually tracked PuckControl, utilising a particular arrangement of IR LEDs to track position and button state. We endeavour to improve upon the PuckControl by; tracking orientation and height of the devices, allowing multiple uniquely identified devices to be used simultaneously, supporting multiple GUI pointers, supporting front or back projected displays, and out-of-reach interactions.

The TableMouse is a constructed from a modified traditional mouse with custom circuitry (Figure 1). The board holding the LEDs is mounted 30mm above the front of the device, giving space for the user to comfortably place their fingers on the buttons and ensuring their hand does not obstruct the LEDs. When raised from the table, users commonly hold the TableMouse at its edges with thumb, 4th and 5th fingers, allowing the 2nd and 3rd fingers to operate the buttons. It is common for users to stabilise the TableMouse with the 3rd finger in situations where greater accuracy is required.

Whereas the PuckControl employs a camera mounted below a back-projected screen to track devices, the TableMouse uses a top-mounted camera. This allows the TableMouse to support back-projected, front-projected and large LCD screens. The camera position enables the device to be accurately tracked in three-dimensions while being lifted from the tabletop surface. Up to nine IR LEDs on the TableMouse are used for the tracking of 3D position, 1D orientation, button state, and unique identification.

Features

The TableMouse is designed to be both a graphical user interface (GUI) pointing device and a tangible interaction device that supports a wide range of tabletop configurations and software. The software layer of the TableMouse consists of 1) a server backend performing the image processing and messaging of TableMouse events to registered client applications and 2) an optional client framework.

Table 1: TableMouse software configuration

	Linux w. MPX			Windows, Mac OS X		
Features	Single cursor	Multi cursor	4DOF	Single cursor	Multi cursor	4DOF
System/OS	•	•		•		
Legacy Apps	•	•		•		
Custom Apps	•	•	•	•	•	•

The TableMouse supports most tabletop configurations and legacy applications currently available. At its most basic configuration, TableMouse can be used as a single operating system cursor device. This provides the TableMouse the ability to seamlessly interact with legacy applications while maintaining orientation independence. Under Linux with the Multi-Pointer X Server (MPX) described below, TableMouse can support multiple pointing devices at a system level and therefore integrates multiple cursor support for legacy applications. Finally, custom applications that support multiple cursors with 4DOF information can use the TableMouse client framework to subscribe and interpret TableMouse server events into unique device information, making use of the full capabilities of the TableMouse. These various configurations are surmised in Table 1.

Orientation independence

Being an absolute positioning device allows the TableMouse to be orientation independent from the display, letting users operate the device from any edge of the screen. To improve usability when operated from an orientation that is not aligned with the system orientation, the TableMouse positions the system cursor in-front of and orientated with the device (Figure 2). This feature is also crucial to ensure the user can aim with the cursor as it renders it always visible.

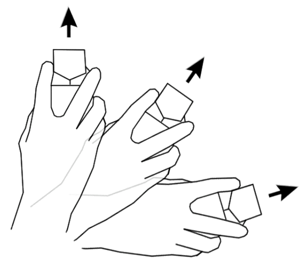

Figure 2: Cursor orientation

4 Degrees of Freedom

Being a visually tracked device allows the TableMouse to perform 4DOF tracking (x, y, z, and θ) in real time. Orientation (θ) is calculated in 1D around the camera's view direction.

TableMouse as a pointing device

The TableMouse can operate as a normal Microsoft Windows, X Windows or MacOS X mouse device. The software to operate the TableMouse as a pointing device consists of a client application called *Squeak*, which listens for TableMouse events and propagates them back to the underlying operating system as native system cursor and button calls. This allows the TableMouse to be used with all legacy applications that are not natively aware of the TableMouse.

Specialised applications can use the TableMouse client application framework to subscribe to the same events as Squeak to determine the orientation and height of the device controlling the system cursor. This allows a user to seamlessly work with legacy applications, where the TableMouse would operate as a traditional 2DOF mouse, and TableMouse aware applications, where the device would have the full 4DOF support. Such functionality is essential for a tabletop device as this reduces cognitive context switching that would otherwise occur when changing from legacy applications to custom applications. Having two different physical input devices for each set of applications would require the user to switch devices.

Unique Identification

Each TableMouse may be uniquely identified through a binary pattern of the fixed IR LEDs. To improve identification under situations where some of the identification LEDs are occluded, a Cartesian tracker has been implemented into the TableMouse software. The tracker uses a set of heuristics to correct any misidentifications and also provides the device position to be estimated when obscured.

Multiple pointer support

The TableMouse support software has the ability to have up to 16 devices operate as 16 independent X Windows system cursors. To support multiple cursors we use MPX (Hutterer and Thomas, 2007), a windowing system that natively supports Single Display Groupware features. Such a GroupWare Windowing System (GWWS) has several advantages over groupware toolkits or groupware applications. Most real-time groupware toolkits only allow one application to be executed at once but a GWWS like MPX has no limitation on how many applications can be executed simultaneously. In general, SDG toolkits exclude legacy applications or do not enhance legacy applications with SDG features. MPX provides groupware features like multiple independent mouse input devices to all applications, regardless of the application's support.

MPX[1] enhances the current X Window System and provides one cursor per connected device as well as annotation overlays and floor control features. The events emitted by MPX are compatible with legacy X Windows applications. The MPX SDG features can be administered in the window manager and are available for both custom-built SDG and legacy applications.

Each TableMouse interfaces with MPX via the TableMouse server using the ImPS/2 protocol and thus presents itself as a regular mouse. When using MPX client applications do not need to register with a TableMouse server directly as events are passed up through the X Windows system. This provides the ability to interact with any application including legacy applications immediately. Multiple TableMouse devices can then interact in several legacy and non-legacy applications simultaneously.

Out of Arms Reach Interaction

While we are primarily interested in investigating an absolute positioning device, being able to interact with parts of the display which are out-of-reach, similar to the pointing feature of TractorBeam (Parker et al., 2005), is desirable. As a by-product of the top-down camera configuration, the TableMouse can operate as an absolute-positing device or as a scaled positioning-device, where movements of the device are scaled onto the display. When operated exclusively on the tabletop surface there is a one-to-one mapping between the cursor position and the position of the device on the display. As the device is raised above the table's surface however, the cross section of the capturing camera's view frustum is reduced and movements of the device are scaled relative to the device's height from the table.

This property allows the user to manipulate cursors in a larger display area than their arm's reach is capable of,

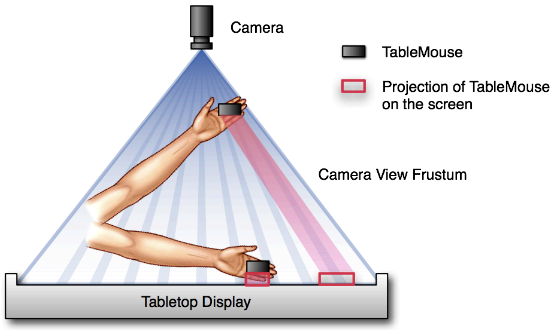

Figure 3: Camera configuration

as depicted in (Figure 3). The black rectangles in the figure indicate positions of TableMouse devices. By raising the device from the tabletop surface the user is able to interact with the far end of the display thus reducing the user fatigue that can occur when they are required to physically reach across the table.

Through user study feedback, we have found this unique out-of-reach interaction to be a beneficial attribute of the technology. One user commented positively that in certain cases the device was less strenuous than the e-Beam. This was especially obvious in cases where the user was not tall enough to comfortably reach the end of the table. Based on this feedback we have implemented a "scaling" property into the TableMouse software, allowing the out-of-arms reach to be exaggerated on a per-user basis.

Out of arms reach interaction is of interest to the territories (Stacey et al., 2004) that occur naturally in collocated collaboration. Users collaborating on a tabletop have tendencies to partition the available space into personal, group and shared territories. Personal territories are established closest to the user for ergonomic reasons, under which case it would be preferential to use the TableMouse exclusively on the tabletop to stabilize the hand for better accuracy. Group territories represent shared space between all users and tend to be located in a position optimal for all those involved. In this case, the out-of-reach interaction would be preferential as the user could reach further into the group territory without having to relocate. Furthermore, collisions in group territories may be avoided as users can "hover" the TableMouse above the other users, an aspect which just is not possible with many touchscreen and pen-based interactions.

[1]http://wiki.x.org/wiki/Development/Documentation/MPX

VISION ANALYSIS PROCESS

The image processing side of the server is built upon the open source OpenCV image processing library. Images are captured from an IEEE1394 camera at 30fps at a resolution of 1024x768, in 8bit mono. Vision analysis is performed on the images to identify the various devices, their positions, and orientations. Being a visually tracked device, the TableMouse's accuracy is restricted by the resolution of the capturing camera. To counter this, all image analysis and image undistortion is calculated at a sub-pixel level.

EXAMPLE APPLICATIONS

We developed two applications to highlight the functionality of the TableMouse; *CheeseDraw* and *MalaMinya*.

CheeseDraw

CheeseDraw is a vector-based drawing application that demonstrates the rich interactions enabled by the TableMouse. CheeseDraw leverages the 4DOF nature of the TableMouse in creating, manipulating and deleting graphical objects on a canvas, while still allowing the user to operate legacy applications.

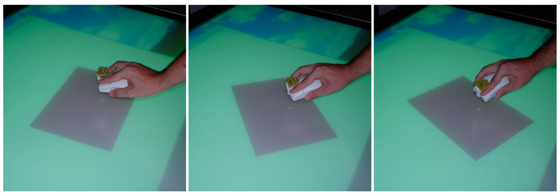

Figure 5: User performing rotation in CheeseDraw

CheeseDraw appears as a standard GUI application with a minimal interface. The left side of the window contains controls for the user to select the type of vector shape they will create, which includes rectangles, ellipses and photographs. The process of creating vector shapes resembles most drawing applications, requiring the user to click and drag regions on the canvas with the TableMouse to specify position and size of a new shape.

Once created, the user may manipulate the shapes using the full capabilities of the TableMouse. Clicking and dragging a shape will translate the position of the shape around the canvas. One-degree orientation of the TableMouse is directly mapped to the orientation of the shape (Figure 5). Raising the TableMouse from the table surface while dragging a shape causes the shape to scale respective to the height of the device from the table. These interactions allows a user to alter three attributes of a shape (position, orientation and scale) in one continuous motion of the TableMouse. Right-clicking an object allows a user to fluidly adjust a shape's colour by controlling its hue, saturation and brightness relative to the TableMouse orientation, height and distance respectively.

In the future we plan to expand CheeseDraw to include system level multi-cursor support available with MPX. This would allow two or more users to manipulate shapes simultaneously, and more interestingly, it would allow a single user with multiple devices to translate, rotate and scale two objects (or the canvas) simultaneously.

MalaMinya

Our drawing tool MalaMinya utilises SDG functionality to provide a drawing canvas that can be operated by up to eight users simultaneously. Figure 4 shows MalaMinya being operated by three different users under MPX. Users have unique coloured icons assigned to their cursors and can draw lines, delete with an eraser or wipe the whole canvas. Initiating a tool only activates the tool for the activating device. Toolbars are aligned around the table (two on each side) for close proximity to the user's physical position, and each user has their own toolbar. The user's toolbar is identified by the user's unique coloured cursor icon, indicated on the left side of the user's tool bar. The toolbars are limited to a single user each using MPX's floor-control mechanism, whereas the various colour buttons are accessible for anyone. The available colours are spread around the drawing canvas, and each user can pick a colour at any time.

Figure 4: MalaMinya

EXPERIMENT

Our initial experiment was a comparison of the TableMouse, conventional mouse and eBeam in tabletop display pointing and selection tasks. The goal of the study was to: 1) make observations regarding the participant's experiences with TableMouse, noting in particular the out-of-reach interaction, 2) determine the accuracy and precision of the TableMouse compared to the other devices, and 3) determine whether there is a measurable difference in speed when performing selection tasks with the three different pointing devices. While this experiment does not evaluate the full range of features of TableMouse, such as multi-user support or 4DOF, we see this experiment as a means to benchmark and receive usability feedback on in the TableMouse.

The experiment conducted closely follows the ISO 9241, Part 9 Draft International Standard (ISO, 2000) to evaluate both *performance* and *comfort* of the devices on the tabletop surface. The experiment presents a series of *tasks*, where the participant's objective in each task is to select a single target presented on screen while maximising speed and accuracy.

Participants

Thirty-two unpaid participants from a computer science department and the general public were involved in the experiment. Of these, there were 20 males and 12 females. The mean age of the participants was 26.6 years (SD 7.61, range 22-59). Participants were required to have normal or corrected vision and to be able to stand in front of the tabletop display.

Devices

The three devices used in the study were 1) a traditional wireless optical mouse, 2) an e-Beam pen device (Luidia Inc.), and 3) the TableMouse. A touchscreen was not part of the formal evaluation as no suitable device the size of our tabletop display was available. The traditional mouse was chosen as a baseline technology. The study used an optical Logitech Cordless Click! Plus wireless mouse. The e-Beam technology was chosen as it has been employed for horizontal displays and ubiquitous workspaces. Guimbretière et al. (Guimbretiere et al., 2001) found that the pen-like devices worked quite well for large wall displays to support collaborative tasks, and pens worked well for selection tasks. It is worth noting that an e-Beam does not provide a means to move the cursor on the display without performing a click. The TableMouse operated with a camera mounted approximately 1.5 metres above the tabletop and only the primary button was required for use.

Tabletop Computer System

The experiment was operated on a CAT (Chen et al., 2006) which employs a back-projected, horizontal tabletop display measuring 1320mm x 1000mm. A traditional PC workstation drives the display running Windows XP at a screen resolution of 1280 x 1024 pixels, equivalent to 24.6 pixels per inch. The e-Beam receiver is mounted on the upper left corner of the display and is calibrated before each session.

Design and Procedure

The experiment was a multi-direction pointing task (2D Fitts discreet task). We employed a previously developed Java-based application that presented a set of tasks in which the participant was required to select a series of filled-circle targets on the display (Zucco et al., 2005).

The experiment was divided into six blocks of 40 tasks per device, with the order of devices randomized within and between participants. This is summarized as:

Tasks	1 – 40 (per block)
Blocks	1 – 18 (6 blocks per device)

Tasks began with an initial target at the centre of the screen with a fixed diameter. Once successfully selected, subsequent target's size, distance, and direction were randomized from the following predetermined set:

Target width	20, 40, 80 pixels
Target Distance	60, 100, 200 pixels
Target Angle	0º, 45º, 90º, 135º, 180º, 225º, 270º, 315º

These variables result in a range of Fitts Index of Difficult (ID) values from 0.8 to 3.4 bits. While tasks were based on the ISO 9241-9:2000(E) standard, circle targets are employed as they provide a more accurate index of difficulty.

Each participant attended one experimental session lasting for less than one hour. The supervisor demonstrated how to use each input device and explained the nature of the study. Participants then performed a training session to become accustomed to the requirements of the experiment.

An instrumented application prompts the participants as to which input devices to use before each task and each task is performed with a single interaction device. After the experiments, the participants fill out a survey detailing their experience with the various devices. This survey is from the ISO 9241-9:2000(E) standard with additional questions to elicit free text comments and a ranking of the input devices.

RESULTS

Results were analyzed using a single-factor repeated measure ANOVA with an alpha level of .05 for all statistical tests. The mean movement time across all blocks for the TableMouse was 1.098 seconds with a standard deviation (sd) of 0.049. The mean movement time of the mouse was 1.017 (sd = 0.06). For the eBeam, the mean movement time was 0.803 (sd = 0.163). These results show a statistically difference in the eBeam ($F_{2,15}$=39.59, $p<0.05$) with it being 37% and 26% faster than the TableMouse and Mouse respectively. When considering only conditions where ID>=3, we see that while the eBeam remains significantly faster than both the TableMouse and mouse ($F_{2,15}$=3.682, $p<0.05$), there is no significant difference between the TableMouse and mouse ($F_{1,10}$=4.964, $p>0.05$). The mean movement times for these conditions are summarized in Table 2.

Table 2: mean movement time for conditions with ID>3

Device	MMT (seconds)	Sd
TableMouse	1.496	0.049
Mouse	1.432	0.095
EBeam	1.203	0.217

Error rates across all conditions were 10.03% (sd = 1.49) for the TableMouse, 6.17% (sd = 1.449) for the mouse and 8.7% (sd = 1.313) for the eBeam. Examining error rate by condition shows the accuracy of the TableMouse improves over the eBeam for larger ID values (Figure 6). These differences are not statistically significant however ($F_{2,24}$=0.55, $p>0.05$). From experiment observations it is clear that this is due to the accuracy of the eBeam device, which suffers from: 1) fine-grained alignment issues with the back-projected tabletop display used in the experiment and 2) the loss of visual tracking of the cursor as is it not tracked while e-Beam is lifted off the table.

The throughput of the TableMouse across all blocks was 1.603 (sd = 0.044). The mouse had a throughput of 1.644 (sd = 0.097) and the eBeam had a throughput of 2.413 (sd = 0.141). While there was a significant positive difference in the eBeam ($F_{2,15}$=103.6, p<0.05), there was no significant difference between the TableMouse and mouse ($F_{1,10}$=0.573, p>0.05).

As many participants would have not used a device like the TableMouse before, we expected it to exhibit some learning effect across blocks. Surprisingly however, Helmert contrasts showed no significant learning effects across blocks for the TableMouse – accuracy across blocks for the TableMouse was consistent.

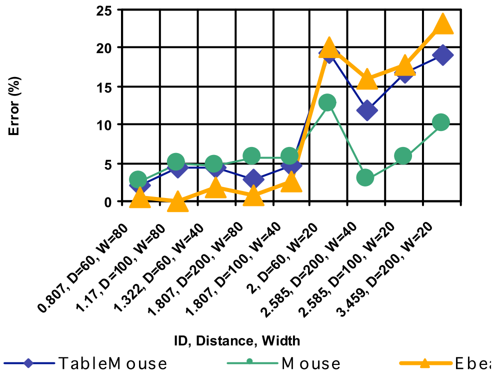

Figure 6: Error rates by ID

Overall, the eBeam had a better throughput than the TableMouse and a comparable throughput to the conventional mouse. Error rates for the TableMouse improved over the eBeam for higher task difficulties, suggesting that the TableMouse is a more accurate pointing device than the eBeam. For accuracy of the large target tasks, we felt all devices performed adequately. The TableMouse has a practical throughput considering the environment for which it specialises (collaborative tabletops).

Survey

The survey provided participants a chance to respond to the usability of the TableMouse on a tabletop display. Some participants reported issues with using a traditional mouse on a tabletop display. They found that the orientation of the screen relative to the mouse made it more difficult to operate than when used on a conventional desktop display. Participants responded positively to the exaggerated out-of-arms reach affect of the TableMouse, as many strained to reach targets at the far side of the display with the eBeam device.

Once participants became accustomed to the differences between the TableMouse and traditional mouse, many made the comment that it felt "intuitive" and that it "[felt] more natural. Less restrictive than the traditional mouse and better able to handle positions further away from where I stood".

The comment that the TableMouse felt "smooth" and "fluid" was made by several participants despite the fact that the device is limited to the camera's frame-rate of 30fps. We believe this perceived smoothness is due to the TableMouse being unrestricted to any surface, unlike a traditional mouse where lifting it from its surface will disrupt the tracking, causing a discontinuity between the cursor and the physical device. This theory is enforced by participant's comments repeatedly mentioning that they appreciated being able to lift the device from the table while still controlling the cursor. As one participant commented, "lifting [the TableMouse] higher from the table increased the speed and accuracy when targeting".

DISCUSSION

The TableMouse represents a new form of interaction device to users whose experience is solely with a relative 2DOF traditional mouse. These users have trained their sensory-motor skills to accommodate the particular attributes of that device's usage. We found that the amount of time required for a user to become accustomed to the TableMouse varied greatly as users recognised how the device was operated and that it was not used like a traditional mouse. The initial difficulties some users had with the device were compounded by the physical shape of the prototype TableMouse, resembling that of a traditional mouse.

Despite the few initial difficulties had by some participants in the study, the potential of the TableMouse in tabletop interaction is worth further exploration. The TableMouse can uniquely and consistently identify each device (and therefore each user) on the table. Being able to consistently track who did what is essential to many CSCW SDG applications as it provides the possibilities of accountability, permission based interaction and personalised interaction. While touchscreens can support multitouch, most cannot uniquely track users in subsequent touches after their hand has left the table's surface. DiamondTouch (Dietz and Leigh, 2001) does support uniquely identified users but is limited to an expensive capacitance based mid-sized tabletop surface.

The TableMouse can provide both absolute position and out-of-reach interaction which, as mentioned previously, benefits users working with group and storage territories in collocated collaboration. Touchscreens require the user to be able to physically reach the part of the screen to interact with it. TractorBeam (Parker et al., 2005) provides a similar "pointing" feature to the TableMouse's out-of-reach interaction but is limited to a single tethered device.

The 4DOF and button states available to the TableMouse can provide rich interaction with applications, as demonstrated by CheeseDraw. While touchscreens can provide up to 4DOF through multitouch interaction (Russell et al., 2005), they do not have a concept of button state. Button state is useful for modal interaction, such as demonstrated in CheeseDraw, where a user can position, rotate and scale or adjust the colour of a shape depending upon which button is being pressed.

CONCLUSION

Through the user study conducted it can be seen that the TableMouse is comparable to other tabletop interaction alternatives in respect to precision and accuracy of selection tasks. We feel this device represents an important low-cost alternative to these devices given the following extra functionality: 1) orientation independence, 2) legacy application support, 3) compatibility with rear and front projected displays (and LCD screens, 4) unique identification of multiple devices, 5) absolute positioning, and 6) precision, 7) 4DOF and 8) out-of-reach interaction. There is no other single device that addresses these features and as such TableMouse is a unique solution.

We have presented the TableMouse and discussed an initial exploration of the aforementioned functionality. CheeseDraw and MalaMinya present two aspects of the TableMouse; 4DOF and multi-user support. The potential of the TableMouse lies in exploiting the TableMouse functionality to its fullest (notably multi-user support). The formal benchmarking against the traditional mouse found the TableMouse was similar in performance and accuracy.

There are a number of future research directions we would like to pursue. We plan to follow with experiment to assess the full functionality of the TableMouse in a collocated collaboration experiment. To improve the feel of the device, we will look at incorporating some features of a pen-like device for an additional pointing device, and we will experiment with homography techniques to provide full 6DOF tracking of the TableMouse, as we envision tilt gestures to be a useful interaction technique.

REFERENCES

Apted, T., Kay, J. and Quigley, A. Tabletop sharing of digital photographs for the elderly. Proc. of the ACM SIGCHI conference (2006).

Benko, H., Wilson, A. D. and Baudisch, P. Precise selection techniques for multi-touch screens. Proc. of the ACM SIGCHI (2006).

Buxton, W., Hill, R. and Rowley, P. Issues and techniques in touch-sensitive tablet input. Proc. of the SIGGRAPH (1985), ACM Press (1985).

Chen, F., Close, B., Eades, P., Epps, J., Hutterer, P., Lichman, S., Takatsuka, M., Thomas, B. and Wu, M.. ViCAT: Visualisation and Interaction on a collaborative access table. IEEE Tabletop (2006).

Dietz, P. and Leigh, D. DiamondTouch: a multi-user touch technology. Proc. UIST 2001, ACM Press (2001).

Guimbretiere, F., Stone, M. and Winograd, T. Fluid interaction with high-resolution wall-size displays. UIST 2001, ACM Press (2001).

Guimbretiere, F. and Winograd, T. FlowMenu: combining command, text, and data entry. Proc. UIST 2000, ACM Press (2000).

Hutterer, P. and Thomas, B. H. Groupware Support in the Windowing System. In Proc. AUIC2007, ACS (2007).

ISO (2000). "Ergonomic requirements for office work with visual display terminals (VDT) - Part 9 - Requirements for non-keyboard input devices."

K. M. Inkpen, M. S. H., R. L. Mandryk, S. D. Scott. Collaboration Around a Tabletop Display: Supporting Interpersonal Interactions. (2001).

Luidia Inc. e-Beam. 125 Shoreway Rd., San Carlos, CA 94070.

Olsen, D. R. and Nielsen, T. Laser pointer interaction. Proc. ACM SIGCHI (2001).

Parker, J. K., Mandryk, R. L. and Inkpen, K. M. TractorBeam: seamless integration of local and remote pointing for tabletop displays. Proc. of GI 2005.

Patten, J., Ishii, H., Hines, J. and Pangaro, G. Sensetable: a wireless object tracking platform for tangible user interfaces. In Proc. SIGCHI 2001, ACM Press (2001).

Patten, J., Recht, B. and Ishii, H. Audiopad: A Tag-based Interface for Musical Performance. NIME 02, (2002).

Potter, R. L., Weldon, L. J. and Shneiderman, B. Improving the accuracy of touch screens: an experimental evaluation of three strategies. In Proc. SIGCHI 1988, ACM Press (1988).

Reitmayr, G., Eade, E. and Drummond, T. Localisation and interaction for augmented maps. In Proc. Fourth IEEE and ACM ISMAR 2005. (2005).

Rekimoto, J. and Ayatsuka, Y. CyberCode: designing augmented reality environments with visual tags. In Proc. DARE 2000, (2000).

Russell, K., Sheelagh, C., Stacey, D. S. and Anthony, T. Fluid integration of rotation and translation. In Proc. SIGCHI 2005. ACM Press (2005).

Ryall, K., Forlines, C., Shen, C., Morris, M. R. and Everitt, K. "Experiences with and observations of direct-touch tabletops." In Proc. IEEE TABLETOP 2006. (2006).

Stacey, D. S., Sheelagh, M., Carpendale, T. and Kori, M. I. Territoriality in collaborative tabletop workspaces. In Proc. CSCW 2004. ACM Press (2004).

Stewart, J., Raybourn, E. M., Bederson, B. and Druin, A. When two hands are better than one: enhancing collaboration using single display groupware. ACM CHI 98. ACM Press (1998).

Takatsuka, M., Lowe, N., Raniga, P. and Quille, M. TR #595: PuckControl: A Computer Vision-based Wireless Input Device for Tabletop Display, School of Information Technologies, Uni. of Sydney (2006).

Underkoffler, J. and Ishii, H. Illuminating Light: An Optical Design Tool with a Luminous-Tangible Interface. In Proc. CHI 1998. ACM Press (1998).

Zucco, J. E., Thomas, B. H. and Grimmer, K. Evaluation of three wearable computer pointing devices for selection tasks. In Proc. ISWC 2005. (2005).