Available online at www.sciencedirect.com
ScienceDirect
Interacting with Computers 19 (2007) 407–427
Interacting with Computers
www.elsevier.com/locate/intcom

User performance with trackball-mice

Poika Isokoski [a,b], Roope Raisamo [a,*], Benoît Martin [b], Grigori Evreinov [a]

[a] *Tampere Unit for Computer-Human Interaction (TAUCHI), Department of Computer Sciences, University of Tampere, Kanslerinrinne 1, FIN-33014 Tampere, Finland*
[b] *LITA, University Paul Verlaine - Metz, ile du Saulcy 57045 METZ CEDEX 1, France*

Received 13 April 2006; received in revised form 21 August 2006; accepted 3 October 2006
Available online 1 December 2006

Abstract

Trackball-mice are devices that include both a trackball and a mouse. In this paper we discuss our experiences in building and testing trackball-mouse prototypes. We report four experiments on user performance with the prototypes used as trackball-mice, conventional mice, and in two-handed configuration with a separate trackball for the non-dominant hand. The results show that user performance with the two-handed configuration was better than in one-handed operation of a trackball-mouse and in one-handed operation of a mouse. Trackball-mouse use and conventional mouse use were more evenly matched. However, Trackball-mouse operation involves a skill that most users do not have whereas mouse operation is familiar to most. Therefore, widespread introduction of trackball-mice does not appear to be justified on performance grounds alone. However, trackball-mice can be used as regular mice by ignoring the ball. This makes them compatible with traditional graphical user interfaces while offering two extra degrees of freedom in tasks where they are beneficial.

Keywords: Trackmouse; Optical mouse; Trackball; Fitts' Law; Pointing device; Two-cursor; Two-handed interaction; Dual-stream input

1. Introduction

HCI researchers are often approached by inventors with ideas on how to improve input devices. One invention that turns up repeatedly is a combination of input devices. Often it is possible to pull out a research paper and explain that the idea has been tested and what the results were. This was not the case with the combination of a trackball and a mouse. We found work on one-handed and two-handed pointing, but only a few papers on one-handed dual-stream pointing, and none on the trackball-mouse combination. Essentially, we were left with theoretical reasons for why use of trackball-mice cannot be as efficient as two-handed use of a mouse and a trackball. We did not, however, have a clear answer to the question of whether a trackball-mouse can outperform the conventional mouse. To clarify this issue we decided to run a series of experiments to investigate user performance with trackball-mice in more detail.

Naturally, interest on trackball-mice is not only motivated by casual inventions. Trackball-mice are already commercially available. This suggests that they have, or are believed to have, value for the users. Available trackball-mice are usually intended for use with standard graphical user interfaces that support only one pointer. If multiple pointing devices are connected to a computer running a typical operating system such as Microsoft Windows, they all control the same pointer. Thus, the utility of typical commercial trackball-mice is limited to the users' choice to decide whether to use the mouse or the ball to move the pointer. This may be practical, for example, when using a trackball-mouse with a mobile computer. The mouse can be used when a suitable tabletop is available, and the ball in other situations.

We wanted to know if there is something more that could be done with these devices. More specifically we were interested in the efficiency of trackball-mouse in combined

* Corresponding author. Tel.: +358 3 3551 7056; fax: +358 3 3551 6070.
E-mail address: Roope.Raisamo@cs.uta.fi (R. Raisamo).

0953-5438/$ - see front matter
doi:10.1016/j.intcom.2006.10.003

use of the two input streams. Two-handed use of two pointing devices is known to be relatively easy and efficient in some tasks. Area-selections using a bounding box and some other multi-degree of freedom tasks have been shown to benefit from two pointing devices. Also, in a two-display setup one could have separate cursors for the displays instead of moving back and forth with only one. Could some of the same benefits be achieved in one-handed use of a pointing device that integrates a ball and a mouse?

Our work is a continuation of a study by Martin and Raisamo (2004). They reported surprisingly positive results for user performance with trackball-mice. We replicated a part of their study under more rigorous data logging and with a different trackball-mouse prototype. We also conducted further experiments and studied different trackball-mouse designs. Thus, the contribution of this paper is a more complete account on user performance with trackball-mice.

We begin by giving a brief overview of what we have learned from prototyping and informal testing of trackball-mice. This information, while possibly not of high scientific rigor, will hopefully be of interest to future builders of similar devices. After describing the device design we begin answering the questions of usefulness and performance of trackball-mice by reviewing previous work in dual-stream pointing and the terminology that we use for talking about it. This introductory material is followed by the description of the new experiments. Finally, we discuss the results to assess the potential value of trackball-mice.

2. Trackball-mouse design

It seems that optimal use of trackball-mice involves combined use of the mouse and trackball functionalities. This is where most of our work is focused. Because the commercially available devices are limited to function in controlling only one stream of pointer input, we began by constructing several new prototypes by physically combining mice and trackballs. These were experimented upon informally and the promising ones were selected for further investigation in the experiments reported below. Some of our prototypes are shown in Fig. 1. For brevity and to differentiate our prototypes from trackball-mice in general, we use the term Trackmouse when referring to the prototypes. In other words, trackball-mouse refers to a device that combines a trackball and a mouse in general, and Trackmouse refers to one of the devices shown in Fig. 1.

Prototype 1 in Fig. 1 is the device that Martin and Raisamo (2004) used in their experiment. It is built on a Logitech WingMan force feedback mouse. This device has a lever system upon which a mouse-shaped handle is attached. The lever system can actuate forces to the handle, but when it does not, the mouse can be used like a conventional mouse except that its range of motion is limited. The device can be turned into a trackball-mouse by replacing the original handle with a trackball (Kensington Orbit 3D in Fig. 1). The WingMan force feedback platform is especially good for prototyping because almost any trackball can be attached without worrying about its internal structure as was necessary in our other prototypes.

Fig. 1. A selection of Trackmouse prototypes. From left to right 1 (WingMan), 2 (Auchan), 3 (TrackMan), and 4 (Marble Mouse).

Prototype number 2 is based on an unbranded product that was available in the French supermarket chain Auchan in 2004. It resembles a conventional mouse in shape and size, but has a small trackball instead of the usual wheel. The main problem with such products is the low quality of the trackball. In general good balls are big and heavy with good ball bearings. A big ball integrates some of the hand tremor and precision bearings allow it to turn smoothly and with little friction. The device that prototype 2 was based on had particularly poor ball bearings because the ball also worked as a button. The bearings needed to allow up-down movement in addition to rotation. Nevertheless, we combined the electronics of two such devices to allow simultaneous use of the ball and the mouse functionality. From this we learned that it is better to take a good trackball as the basis of the prototype and then find a way to embed a mouse in it rather than to start with a mouse-like body.

In prototypes 3 and 4 we used Logitech trackball products as the starting point. Prototype number 3 was built using a TrackMan Wheel trackball, and prototype number 4 using a Marble Mouse trackball. We tried to integrate different optical mice within them, but ended up using the Mitsumi FreeStyle mice because of their compact electronics based on HDNS-2000 sensor from Agilent Technologies. Embedding the mouse optics is the most demanding phase of the construction. The distance from the surface below the mouse needs to be just right (for the HDNS-2000 between 2.3 and 2.5 mm). The mouse optics have a very narrow focus range and the surface tracking algorithm performance suffers if the mouse cannot see the surface clearly. Apart from mounting the optics the construction is a simple matter of cutting away some extra plastic within the trackball, shaping the aperture for the optics in the bottom plate of the device, and routing the mouse cord out safely without disturbing other mechanics such as the wheel and buttons. Special care must also be taken to anchor the optical elements well. They should not move at all in relation to the aperture, or else the mouse performance may suffer. The prototypes used in the experiments were tested on various surfaces to make sure that the mouse functionality did not suffer in comparison to the original mouse. The rubber pads under the device must be removed to make it slide better. Slippery pads can be used as replacement, but we found that just letting the device slide on its bottom plate works well enough on the surfaces that we planned to use.

Overall, our prototyping resulted in some devices that we felt were quite usable. However, in addition to a good ball, a trackball-mouse needs a good cover design that allows a comfortable grip in combined trackball and mouse use. We limited our exploration to the trackball covers that were available in the nearby computer shops. Better shapes may be possible with a complete chassis re-design.

To experiment with two separate pointing devices in Microsoft Windows environment, we needed a mouse driver that unlike the standard driver allowed us to separate the events generated by the two pointing devices in the Trackmouse. We modified the CPN[1] driver originally written by Westergaard (2002) to suit our purposes. The result was that our experimental software received ordinary mouse events from the mouse part of a Trackmouse, and special CPN-events from the trackball-part. Obviously a standard solution for delivering multiple-pointer data to applications is needed. We do not propose that ours is ideal as a generic solution. However, it was sufficient for research purposes.

3. Dual-stream input

In this section we discuss dual-stream input with the two input streams generated with one or two devices operated by one or two hands. To avoid complications we define an input stream to be what one typically receives from a pointing device such as a mouse. So, an input stream in the following discussion includes two-dimensional motion data and button events. The goal in the following discussion is to explain the terminology that is used in this paper and to clarify our choice of tasks to test, not to give a complete taxonomy of input modes.

3.1. Classification of dual-stream inputs

Since dual-stream input can be considered to be a special case of multimodal interaction, we utilize a well-known classification of multimodal user interfaces as a basis of our terminology. Nigay and Coutaz (1993) presented a classification for multimodal user interfaces defining the two main dimensions: use of modalities (sequential, parallel) and fusion (independent, combined). We present an extension of this classification to describe the different types of dual-stream input. Fig. 2 shows the extended framework. The numbers in Fig. 2 refer to the experiments discussed later in this paper. We have divided the categories of Nigay and Coutaz in one-handed and two-handed subcategories. This approach also applies to Chatty's (1994) classification of two-handed interfaces.

When only one pointing device is used in multiple operating modes, the modes are always combined sequentially. This configuration has only a single-stream and is not included in Fig. 2. The interpretation of other possible configurations is as follows.

The *use of streams* can be sequential or parallel. Dual-stream parallelism can exist either between hands or within a hand.

When two streams are used for doing two unrelated things, their use is *independent*. In *exclusive* tasks the two streams are used one at a time. The main problems in this kind of dual-stream input are related to motor differences between the hands (two-handed case), and to the overloading of the one hand with too many functions (one-handed case). In *concurrent* tasks the streams are used at the same time, but since the tasks are independent they require divided attention making these tasks cognitively difficult. However, such tasks do appear in special situations such as games and simulators that require handling several parallel actions. Concurrent two-handed input requires user training to be useful but sometimes the training is worth the

	USE OF STREAMS			
FUSION	Sequential		Parallel	
COM-BINED	ALTERNATE		SYNERGISTIC	
	One hand (1, 3, 4)	Two hands (3)	One hand (2)	Two hands (2)
INDEPEN-DENT	EXCLUSIVE		CONCURRENT	
	One hand	Two hands	One hand	Two hands

Fig. 2. Classification of dual-stream input. The numbers indicate the experiments reported in this paper.

[1] CPN stands for Coloured Petri Nets. The driver was originally written for the use of the University of Aarhus CPN group.

trouble, for example, when driving a car. Concurrent dual-stream input with one hand could happen, for example, with game controllers that may require parallel use of two streams with one hand. The training needs are similar to the two-handed concurrent input. In this paper we focus on tasks where it is preferred to avoid divided attention. Thus, exclusive and concurrent tasks receive little attention in the remainder of this paper.

Two-handed interaction works best when the parallel streams of input are *combined* to accomplish a common goal (Chatty, 1994; Nigay and Coutaz, 1993). This kind of input has potential for giving the user a cognitive advantage over single-stream interaction (Leganchuk et al., 1998) by allowing the user to act in a more natural level of chunking the task. If the use of different streams is sequential, combined interaction is *alternate*. In this case a combined input is formed of a sequence of inputs that add up to a common goal. One- and two-handed alternate tasks differ in motorics since using one hand in sequentially combined tasks involves transitioning the same hand from a task to another, whereas in the two-handed case the hands may maintain optimal working posture when idle. Alternate multi-stream tasks may also have some level of cognitive benefit over single-stream tasks. *Synergistic* tasks are like alternate tasks, but involve parallel use of input streams. Two-handed tasks that have shown the largest improvements in execution speed and naturalness over their single-stream counterparts have been synergistic (Buxton and Myers, 1986; Kabbash et al., 1994; Leganchuk et al., 1998; Hinckley et al., 2002; Zhai et al., 1997; Latulipe et al., 2005a,b, 2006.). The motor challenges in one-handed synergistic use are high since the mechanics and neural control of the hand do not allow movements to be combined as freely as it is possible with two hands.

In our experiments we concentrated on tasks where two input streams are combined in an alternate or in a synergistic way. Within this domain there is still a further classification criterion. Namely, since the dominant and non-dominant hands have different accuracy (Annett et al., 1979; Flowers, 1975; Schmidt et al., 1979; Todor and Doane, 1978), it is often useful to assign different roles to different hands (Guiard, 1987). If this is done, the interaction is *asymmetric*. If, however, both hands have the same role, the two-handed interaction is said to be *symmetric*.

Asymmetric two-handed interaction is common in the real world and has also produced good results in user interfaces. Many tasks in the earlier experiments have been asymmetric in nature, following Guiard's (1987) kinematic chain model. The model states that the non-dominant hand acts first to set up a context for the dominant hand that terminates the action. The dominant hand acts relative to the non-dominant hand and has a finer spatiotemporal scale than the non-dominant hand.

A common example of *symmetric* two-handed interaction is typing on a QWERTY keyboard: each finger performs a discrete task of pressing a key, and there is no difference between the hands in the way that they are used.

We use these terms in the context of one-handed dual-stream input because the different streams have different properties leading to asymmetry similar to two-handed input. For example, a trackball is less efficient than a mouse. This could lead to optimal use of the input streams of a trackball-mouse in roles similar to those of the dominant and non-dominant hands.

3.2. Studies on two-handed interaction

Ever since Ivan Sutherland's Sketchpad system (Sutherland, 1963) two-handed interaction has existed in user interfaces. In sketchpad the light-pen and rotating knobs could be used to move and scale or rotate objects simultaneously. The currently ubiquitous combined use of a keyboard and a two-dimensional pointer was also present: the non-dominant hand was used to choose the operating mode using a matrix of buttons while the task itself was carried out with the dominant hand holding the light-pen. Today practically all graphical user interfaces make use of modifier keys (control, alt, shift) to alter the meaning of mouse operations.

Krueger (1983; Krueger et al. 1985) suggested the use of two hands to give continuous input in a coordinated and integrated manner. He proposed two-handed gestures that could be used in text editing and two- and three-dimensional graphics. Buxton and Myers (1986) presented the empirical data on two-handed interaction in desktop systems. They reported two kinds of benefits: naturalness and efficiency. Two-handed interaction has indeed proved to be natural and efficient in comparison to conventional use of one pointing device, at least in some classes of tasks (Buxton and Myers, 1986; Hinckley et al., 1998; Kabbash et al., 1994; Zhai et al., 1997; Leganchuk et al., 1998; Martin and Raisamo, 2004; Latulipe et al., 2006). This is mostly due to cognitive benefits due to better chunking of task components and because it is mechanically easier to operate two relatively independent hands in simple ways than one in a complicated way (Leganchuk et al., 1998).

Symmetric bimanual interaction with pointing devices has been studied less than asymmetric interaction. Balakrishnan and Hinckley (2000) found that visual integration of the pointers is necessary for efficiency in symmetric tasks, such as two-handed rectangle editing adjusting the opposite corners of the rectangle as done by Casalta et al. (1999), Leganchuk et al. (1998) and Latulipe et al. (2005b). Two-handed rectangle or ellipse editing is a popular task in bimanual studies. Buxton and Myers (1986) used an asymmetric version. A symmetric "rubber band" version was used by Casalta et al. (1999). Leganchuk et al. (1998) used a similar task where objects were to be enclosed by a minimal rectangle or ellipse. Latulipe et al. (2005b) studied image alignment, but the image they aligned was rectangular.

Another commonly reported task type is a two-handed pointing task like, for example, the connect-the-dots task by Dillon et al. (1990) and Kabbash et al. (1994). Kabbash et al. had also asymmetric bimanual conditions while Dillon et al. varied the size of the menu for the non-dominant hand and had different selection modalities (touch and speech). Dillon et al. (1990) found no difference between one- and two-handed performance. Kabbash et al. (1994) found that the asymmetric methods performed best, but the symmetric two-handed control of two pointers may be even slower than a conventional one-pointer mouse.

Latulipe et al. (2006) showed that a symmetric two-handed technique for manipulating b-splines can outperform one-handed and asymmetric two-handed techniques.

3.3. Dual-stream input with one hand

We continued the work of Martin and Raisamo (2004) who reported results on the use of two streams of continuous 2D input using only one hand and a Trackmouse. There are four other one-handed multi-stream studies. The devices used in these studies were modified mice integrating a wheel, an isometric joystick, a touchpad, and a tilt sensor with the mouse.

Zhai et al. (1997) compared a standard mouse, a mouse with a wheel (Microsoft WheelMouse), a mouse with an isometric joystick (IBM TrackPoint III) and a two-handed mouse-joystick setup in a scrolling task. The results showed that the two-handed technique was the fastest, but the mouse with an isometric joystick was almost as fast. However, the mouse with a wheel was even slower than the standard mouse. Since there were some issues concerning how the wheel was used, Hinckley et al. (2002) repeated the experiment varying the scrolling distance. They showed that in dual-stream interaction the IBM ScrollPoint mouse performed best at long distances, but the Microsoft IntelliMouse Wheel performed best at short distances. In general, these studies show that both devices are usable in dual-stream interaction. However, in comparison to the trackball-mice we must notice that the second input stream in these experiments was one-dimensional.

Balakrishnan et al. (1997) concentrated on 3D manipulation with a mouse that could be tilted and Balakrishnan and Patel (1998) on marking menu use with a touchpad integrated in a mouse. Both observed good performance and concluded that devices integrating more degrees of freedom in a one-handed device can be useful.

4. Overview of the experimental program

Our research question is open-ended: What are the characteristics of user performance with a trackball-mouse? This makes answering it complicated. Different ways of using the pointer data in applications result in different demands for visual, motor, and cognitive effort. There is no single task that can be used to elicit all the relevant data. Therefore, a series of experiments exploring the different cells in the conceptual framework in Fig. 2 is needed to understand the characteristics of a multi-stream input device. Below we report a series of four experiments.

We utilized some tasks found in the literature. However, we also wanted to explore new tasks. In other words we feared that some results may be more task-specific than general to the input device configurations being studied. We studied user performance with a trackball-mouse in fairly generic pointing tasks. We preferred tasks that occur regularly in the use of normal graphical user interfaces even though we know that using complicated multi-degree-of-freedom tasks would have made trackball-mice look better. Cognitively simple tasks allowed us to focus on the basic motor capabilities of the users and register data on user performance in the simple building blocks of pointing device usage that are needed for more complicated tasks.

In the first experiment we compared user performance with the two Trackmouse prototypes that we considered the best based on our informal testing. The winner of this comparison was used as a representative of trackball-mice in the other experiments. The second experiment repeated the ellipse task (task 2) of Martin and Raisamo (2004) to see if our prototype would produce the same promising results that they reported. After this, we continued with experiments three and four that explored the use of a second cursor to operate a toolbar in a simulation of a pixel manipulation program. The third experiment was concerned with one- and two-handed novice performance, and the fourth was a longitudinal study aiming to measure whether the Trackmouse usage skill develops as expected.

5. Experiment 1

Our first task was to select a good Trackmouse prototype to use in further experiments. We wanted a prototype without the limited motion of the Logitech force-feedback platform, and with a reliable and easy to operate ball. These criteria still left two prototypes. One with a thumb-operated ball and one with a finger operated ball. In the first experiment we compared these two prototypes in three conditions: (1) the mouse used alone, (2) the ball used alone, and (3) sequentially and symmetrically combined use of the mouse and ball input.

5.1. Participants

Twelve participants (9 male, 3 female) who had not participated in the other Trackmouse experiments were recruited from the staff of University of Tampere. The mean age of the participants was 29 years (range 23–43). All participants were right-handed and used computers daily in their work, but did not use trackballs regularly. One participant reported substantial trackball usage history. He had had access to a laptop with a small trackball that could be attached to the side of the laptop. The laptop in question had fallen to disuse during the last 18 months prior to the experiment, but the usage skills had probably not been forgotten. In this and the other experiments reported in this paper the participants were volunteers, they were not compensated for participating except for being allowed to participate during working hours.

5.2. Apparatus

An 800 MHz Intel Pentium computer with Windows XP was used. The experimental software presented the task and saved the user's input with timestamps. The display was a 19-inch CRT with 1280×1024 pixel resolution. Participants were seated in a regular office workstation with an adjustable desk and chair. Experiments 1, 2, and 3 utilized the same computer and office. The mouse settings were also the same. The mouse sensitivity was set to the middle position and acceleration was turned off.

Prototypes 3 and 4 in Fig. 1 were used. The trackball part of these devices was different, but the mouse part was identical. Both had an optical Mitsumi Freestyle mouse embedded within.

In this and the following experiments the rightmost button of the device was associated with the cursor controlled with the ball of the Trackmice. The leftmost button was always associated with the cursor that was controlled with the mouse functionality of the Trackmice.

5.3. Task

The task was an adaptation of the multi-directional pointing task of ISO-9241 part 9 Annex B Section 6.6.2. Two screenshots of the starting state of the task display with a 22 cm circle of 30 one-centimeter targets are shown in Fig. 3. Note that the arrows in Fig. 3 were drawn to illustrate the pointer movements; they were not a part of the task display. The task was performed in three different ways: with only the mouse functionality, with only the trackball functionality, and with two cursors using the mouse and the trackball functionality of the Trackmice. The mouse and the trackball use proceeded as suggested in the above mentioned standard (left in Fig. 3). In the two-pointer use the participants were instructed not to move the cursors across the circle, but to alternate between the cursors as shown in the right side of Fig. 3. Thus the long movement across the circle was not needed, but the participants had to constantly switch their attention from one cursor to the other. The target to select was highlighted. In the two-pointer condition two highlighting colors were used to avoid confusion. Light blue was for the cursor associated with the trackball (blue cursor) and white was for the mouse cursor (white cursor).

To select a target the participant pressed a button on the pointing device. Button-down events were used for recording the cursor position. The task proceeded to the next target on every button press regardless of whether the cursor was on a target or not. This is important for smooth progress in the task. Errors must not interrupt the flow of pointing. The coordinates of the cursors at the time of button presses were saved and thus errors could be tabulated in the data analysis.

5.4. Procedure

Participants completed two sessions. Each session consisted of six blocks of tasks completed in three device configurations (trackball, mouse, and Trackmouse) with two devices (TrackMan and Marble Mouse). Each block consisted of eight tasks. Four different target sizes (5, 10, 20, or 40 mm) were used. The diameter of the circle onto which the targets were placed was always 220 mm. These four tasks were circled in clockwise and counter-clockwise directions making the total number of tasks eight per block. In total, each participant completed $2 \times 6 \times 8 = 96$ tasks with 30 targets to click in each (i.e., a total of 2880 targets per participant). Completing all blocks took about 45 min per session.

Including the instructions in the beginning of the first session and filling in the subjective evaluation form and debriefing in the end of the second session the experiment required about two hours of participants' time. The sessions were scheduled on consecutive days or with one day in between.

The subjective evaluation form was a modified version of the form in Table C 1 in ISO-9421-9 Annex C. We removed questions 1–3 and 6 that in our experience tend to produce redundant results. There was only one of each question per device. We asked the participants to give an

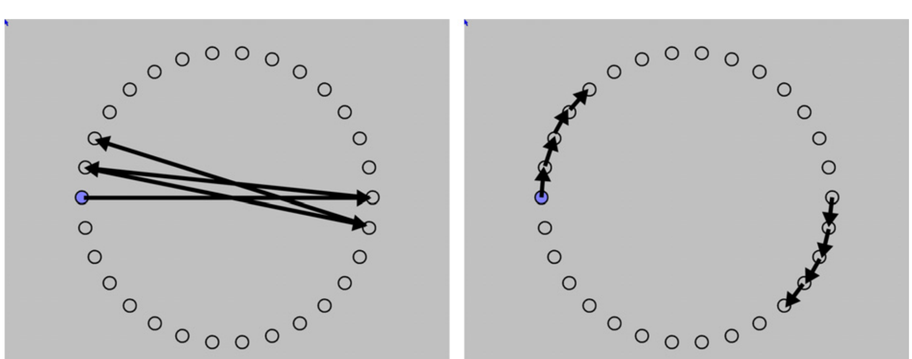

Fig. 3. Cursor movements in one-pointer (left) and two-pointer use (right).

average of their response over the three configurations (mouse, ball, and trackball-mouse). Answers were given by marking one box out of seven ranging from 1 (negative) to 7 (positive).

The participants were not allowed to practice Trackmouse use before beginning the first task. The goal was to record even the very first attempts in using the devices. The purpose of the second session was to record performance after some practice.

When performing the tasks, the participants were instructed to proceed as fast as possible while making as few errors as possible. One or two errors per task was instructed to be ideal. The software played a sound when the participant clicked outside the target making it easy for the participant to notice errors.

It was possible to make different kinds of errors. The user might use the correct cursor and the correct button but miss the target with the cursor. This kind of miss is dealt with in the Fitts' law based analysis. Larger spread of pointing coordinates is penalized by a larger effective target width in the throughput calculation. However, there were other kinds of errors that must be dealt with by other means. The user might click the wrong button, which means that a target selection event occurs at the location of the other cursor (which may be anywhere, but most likely on the other side of the display). It is also possible to double-click when selecting a target. This also leads to a target selection on the wrong side of the display because the second click is associated with the next target. We excluded these anomalous data points from our analysis by removing all data points where the button was clicked further than 110 mm. from the highlighted target. In total, 0.3% of the data was excluded from the Fitts' law based analyses. However, all data were included in the error rates reported below.

Button-down events were used for timing. Time for a target began at the button-down event on the previous target and ended on the button-down event associated with the current target.

5.5. Results

5.5.1. Statistical tests

Throughout this paper, the results of the repeated measures ANOVAs have been verified to hold under Greenhouse-Geisser correction for sphericity violations. For simplicity we report the results with the traditional un-corrected degrees of freedom. Error rate data were log-transformed before testing. Other data were not. Holm's procedure was used for controlling family-wise error rate when doing multiple pair-wise t-tests. The subjective ratings were compared using Wilcoxon's signed rank tests for paired samples.

5.5.2. Throughput

As our main criterion in the comparison of the devices we use the throughput measure as defined in the ISO-9241-9 Annex B. We explain this measure only briefly. For more complete description one should refer to the standard (ISO, 2000) and to the more detailed discussion and instructions given by Douglas et al. (1999) and by Soukoreff and MacKenzie (2004).

In short, the data were processed as follows. First, the distances of the cursor from the center of the targets at the times of button presses were used to compute the effective target width (W_e) using the formula

$$W_e = 4.133\sqrt{\frac{\sum_{i=1}^{n}[(x_i-\bar{x})^2+(y_i-\bar{y})^2]}{n-1}}$$

where n is the number of clicks per task, x is the x-coordinate of a click, y the y-coordinate of a click. The effective target width was then used to compute the index of difficulty for each task using the formula

$$\mathrm{ID}_e = \log_2\left(\frac{A}{W_e}+1\right)$$

where A is the distance between consecutive targets. The index of difficulty was then used to compute the throughput for each task using the formula

$$TP = \frac{\mathrm{ID}_e}{t}$$

where t is the mean time per click when performing the task. Throughput was computed for each instance of a task separately and then averaged to get the mean throughput per participant. These per-participant mean values were used in the statistical tests reported below. This procedure may seem complicated considering that we could have just used the pointing time and error rate measures for the comparison. The advantage of using throughput is that the computation for effective target width partially compensates for different speed-accuracy trade-offs that the participants may exhibit. Thus, we get one number per participant that incorporates both speed and accuracy. When viewed alone, the numbers for speed and accuracy tend to vary between tasks and participants. Throughput values are less variable because some of the variability due to varying speed-accuracy emphasis has been removed. This helps in finding statistically significant differences in the results.

Other factors being equal, smaller effective target width produces higher throughput values and larger effective target width produces lower throughput values. Ideally this means that a fast but sloppy participant and a slow but accurate participant tend to produce the same throughput. Details of the proper data-analysis procedure are still being debated. See for example Soukoreff and MacKenzie (2004) and Zhai (2004). However, even if the throughput computation does not compensate for the speed-accuracy tradeoff completely, and does not completely guarantee the inter-study comparability of results, it is a valid measure in within-study comparisons as long as the A/W conditions presented in the compared situations are the same and the intercepts of the regression lines (shown in Fig. 4) are close to each other. In this case it appears that these assumptions are not violated.

The throughputs for all device conditions in both sessions are shown in Fig. 5. A $2 \times 2 \times 3$ (session $\times$ device $\times$ device configuration) repeated measures ANOVA confirms what can be seen in the Fig. 5. Namely, that all three factors have a statistically significant main effect (session: $F_{1,11} = 140$, $p < 0.001$, device: $F_{1,11} = 8.8$, $p < 0.05$, and device configuration: $F_{2,22} = 197$, $p < 0.001$). Additionally there is a statistically significant session by device configuration interaction ($F_{2,22} = 23$, $p < 0.01$) meaning that the effect of training was different in different device configurations. Looking at Fig. 5 it appears that the combined use of the mouse and trackball functionality shows the largest improvement between sessions 1 and 2. Also, the device by device configuration interaction ($F_{2,22} = 11$, $p < 0.01$) was statistically significant meaning that different configurations performed differently on different devices. This suggests an answer to our main question of which of the two devices was better. A closer look at the results for the second session in Fig. 5 reveals that there was no difference in throughput between the prototypes when both the mouse and the trackball were used to complete the task ($t_{11} = 1$, $p = 0.36$). Also, the trackballs when used alone resulted in almost equal throughput ($t_{11} = -0.38$, $p = 0.71$). The only statistically significant difference between the devices was found when only their mouse functionality was used ($t_{11} = 4.7$, $p < 0.01$). In this case, the TrackMan prototype had significantly higher throughput (5.4 vs. 4.7 bps). Error bars in Fig. 5, and when present in all figures in this paper, show the standard error of the mean.

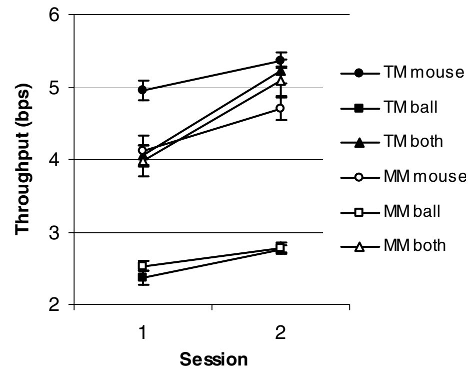

Fig. 5. Throughput: TM is for TrackMan, MM is for Marble Mouse.

For easy comparability with the one-pointer configurations the throughput for the two-pointer configuration was computed using the same distance ($A = 220$ mm) as in the one pointer conditions. This is the distance between the targets, but does not correspond to the actual pointer movements (as seen in Fig. 3). Figs. 4 and 5 are good for comparing the one- and two-pointer configurations in this experiment. However, for future cross-study comparisons, the throughput computed using the actual pointer movement distance ($A = 23$ mm) may be better. In the second session these throughputs were 2.12 bps for the TrackMan and 2.04 bps for the Marble Mouse prototype.

Two observations should be made on the regression lines shown in Fig. 4. Firstly, the experiment included only variation in target width (W). The distance between the targets (A) was always the same. Thus, these results may not be comparable with results of experiments that have manipulated both A and W. Secondly, all lines have a negative intercept. The negative intercepts were smaller for the second session possibly indicating that whatever the anomaly causing them was, it seemed to be getting smaller with training. If we were to make conclusions based on the regression equations, the only difference to what was said above would be that in the combined configuration the Marble Mouse prototype performs slightly better than the TrackMan prototype because it produces a lower slope. This seems to be mostly due to the two highest points in the "both" series in Fig. 4. These points correspond to the smallest (5 mm) targets. This agrees with the difficulties in precise operation of the TrackMan ball that some participants mentioned. This disadvantage should be weighted against its superior mouse-only throughput, the error rate results, and the subjective ratings reported below. It should also be noted that only the mouse conditions had the same

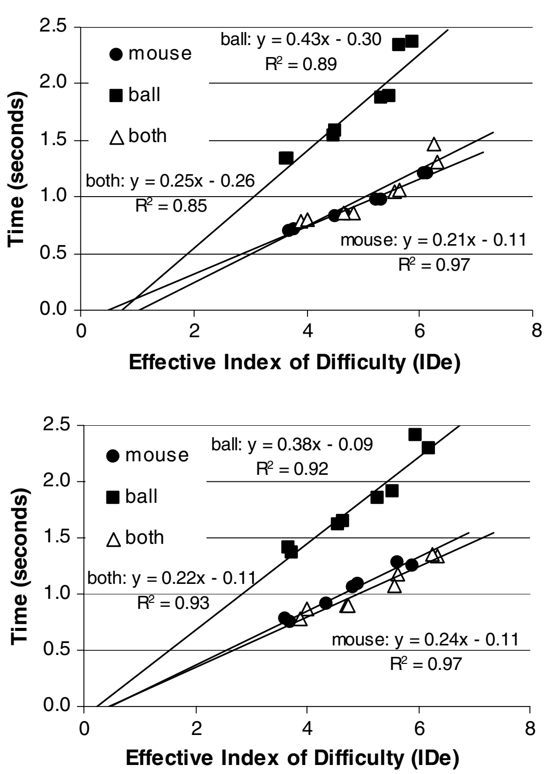

Fig. 4. Fitts' law regression lines for the TrackMan prototype (top) and Marble Mouse prototype (bottom) based on data from the second session.

intercept. Other slope comparisons must consider the differences in intercepts as well.

5.5.3. Error rate and pointing time

The Fitts' law analysis above gave results where pointing time and error rate were combined. We omit the detailed results of separate pointing time and error rate analysis except for a few main points that help to explain some of the findings reported above.

The overall average error rate was 2.2% in the first session and 1.3% in the second session. The only device configuration that was visually, but not statistically significantly, different from the others was the TrackMan prototype used as a mouse only. The error rate with it was as low as 0.5% in the first session and 0.4% in the second session. This amplifies the effect of the lowest pointing time to produce the highest throughput. Overall, the error rate distributions were highly skewed and the data had a high variance. No statistically significant differences were found.

5.5.4. Estimating the cognitive cost of trackball-mouse operation

Because we now know the throughput that users achieve with the mouse and trackball components of the Trackmice, we can compute an estimate for the time needed for the motor activity when completing the task with a Trackmouse and subtract that from the measured pointing time to see how much extra time the participants spent when operating the trackball-mouse in the two-pointer condition. This extra time is likely to be associated with the cognitive cost of switching between the mouse and ball functionality. The difference is 667 ms for the TrackMan prototype and 707 ms for the Marble Mouse prototype in the first session and 418 and 413 ms, respectively, in the second session. It seems that this extra time (cognitive cost) is going down with practice, but our data does not allow reliable extrapolation to estimate how far it will decrease. The level of this difference that can be reached with continuing training is critical regarding the usefulness of trackball-mice in symmetric and alternate tasks. Experiment 4 investigated the effect of training more thoroughly.

5.5.5. Subjective ratings

We found no statistically significant differences in the subjective ratings. The only question that elicited a marginally significant difference ($z = 1.9$, $p = 0.06$) was the question on "overall operation of the device" in which the TrackMan prototype was rated higher.

5.5.6. Summary

The only statistically significant difference between the devices was found when the devices were used as mice only. In this case the TrackMan prototype (prototype 3 in Fig. 1) had a higher throughput. Based on this, we concluded that the TrackMan prototype was the better one, and used it in the following experiments.

6. Experiment 2

Our second experiment was a replication of task 2 in the experiment by Martin and Raisamo (2004). Martin and Raisamo found that a Trackmouse can outperform a conventional mouse in ellipse resizing and positioning tasks. Their experiment, however, was done with an early Trackmouse prototype that was built on a force-feedback mouse with potentially abnormal ergonomics (prototype 1 in Fig. 1). Before extending their work we wanted to confirm the findings using the same devices that would be used in our further experiments.

6.1. Participants

Twelve employees (7 male, 5 female) of the University of Tampere participated in the experiment. The mean age of the participants was 29 years (range 24–45). All were right-handed mouse users. One participant was left-handed, but had always used the mouse with his right hand. Some participants had occasionally used trackballs and two-handed user interfaces in demonstrations of such systems, but none had used the two-handed device configuration that we had in this experiment. None of the participants had used a Trackmouse before.

6.2. Apparatus

The apparatus used in this experiment was the same as in experiment 1 except for two differences. Firstly, the input devices were different. The Trackmouse used in this experiment was the prototype number 3 in Fig. 1 and the trackball used with the left hand in the two-handed setup was a MacAlly iballpro which is a sturdy desktop trackball with a 5 cm ball. Secondly, the software that presented the task and logged user input was different.

It would have been desirable to use the same trackball in the Trackmouse and two-handed configurations. However, the TrackMan trackball used for the Trackmouse prototype that was found superior in experiment 1 had an asymmetric shape and, therefore, it was unsuitable for left-handed use.

6.3. Task

The task was to surround a group of dots with a white ellipse as shown in Fig. 6. In the experiment the dots to surround were red. In Fig. 6 they are shown in white to ensure their visibility on grayscale printouts. Initially the ellipse was round and its size was about halfway between the smallest and largest ellipse needed in the tasks. The task window filled the whole display. The red dots were placed so that it was always possible to surround them with an

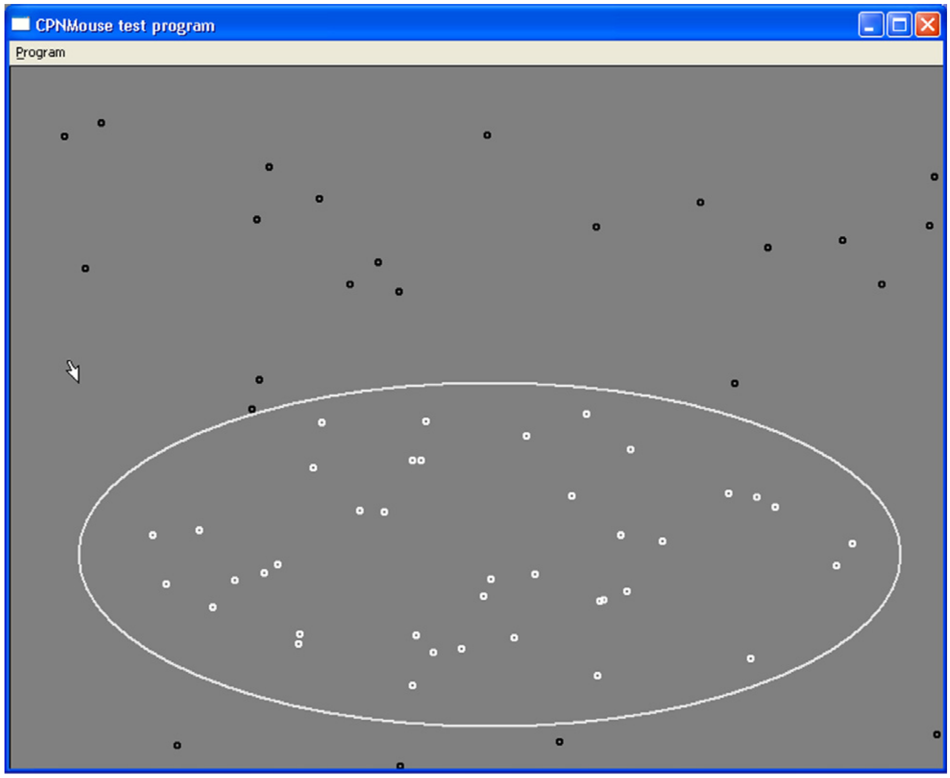

Fig. 6. The ellipse task.

ellipse without including any of the black dots that populated the rest of the area.

The task was completed with three different input device configurations. We call these the mouse, the Trackmouse, and the two-handed configuration. In all cases the Trackmouse was used with the right hand. In the mouse and two-handed configurations only the mouse functionality of the Trackmouse was needed.

In all device conditions the control of the ellipse happened through cursors that were anchored onto the center and the upper-left corner of the (invisible) bounding box of the ellipse. The center cursor moved the ellipse and the upper left corner moved in relation to the center to change the shape and size of the ellipse. The position of the upper left corner was relative to the center so that when the ellipse was moved, it retained its shape and size. The upper left corner was mapped to the ball and the center to the mouse except in the mouse-only mode where the user needed to press the right mouse button to switch between controlling the center and the bounding box. To make it easier to differentiate between the cursors, the cursor associated with the upper left corner was drawn upside-down as shown in Fig. 6. In the mouse mode only the active cursor was shown.

When the participant pressed the left mouse button to finish a task, the ellipse turned black, timing for the task ended, and there was a pause. The participant ended the pause by pressing the left mouse button again. The next task was displayed immediately, and its timing was started.

6.4. Procedure

The experiment was a three device configuration by two block design with the order of the device configurations balanced between subjects. Since there were $2 \times 3 = 6$ blocks, and each block consisted of 25 trials, each participant completed $6 \times 25 = 150$ trials.

The task was explained to the participants while the first task was visible on the display. The participants were instructed to perform the trials as fast and as accurately as possible. The first block with each device configuration began without any practice trials. The intention was to measure user performance during the first few minutes of use. To ease anxiety of performing a new task without training, we referred to the first block with each device configuration as training in the presence of the participants. They were aware of that data on all blocks were recorded.

At the end of the experiment we asked the participants to fill in a subjective evaluation questionnaire. This questionnaire was again an adaptation of the rating scale suggested in Annex C of Part 9 of the ISO-9241 standard (ISO, 2000). It consisted of 10 questions where participants were asked to rate their impressions regarding each device configuration on a scale from 1 (negative) to 7 (positive). Annex C suggests 12 questions. We considered the last three questions (arm fatigue, shoulder fatigue, and neck fatigue) difficult to assess based on the short experiment and replaced them with just one question asking how much the participants liked the device. The participants needed about 45 min to complete the experiment.

6.5. Results

6.5.1. Task completion time

The results on task completion time are summarized in Fig. 7. As seen in Fig. 7, both the device configuration ($F_{2,22} = 17.3$, $p < 0.01$) and block ($F_{1,11} = 25.0$, $p < 0.01$) affected the task completion time. Pair-wise comparisons show that in the first block the two-handed interface was faster than the mouse ($t_{11} = 4.1$, $p < 0.01$) and faster than the Trackmouse ($t_{11} = 3.5$, $p < 0.01$), and that the Trackmouse was faster than the mouse ($t_{11} = 2.8$, $p < 0.05$). In the second block the two-handed interface was again faster than the mouse ($t_{11} = 4.4$, $p < 0.01$) and faster than the Trackmouse ($t_{11} = 5.4$, $p < 0.01$), but the difference between the Trackmouse and mouse was not statistically significant ($t_{11} = 2.1$, $p = 0.06$).

The performance of one of the participants was anomalous in the second mouse block. The average task duration was 14.3 s while the average of the other participants was 8.2 s. However, other than the long duration of the tasks with the mouse configuration, we observed no anomalies. Therefore, we deemed the performance of this participant just as valid as that of the others. Removing the data of this participant would have made the difference between the Trackmouse and mouse configurations smaller, but statistically significant. However, the trend of the mouse configuration catching up to the Trackmouse configuration during the second session would have appeared even stronger in Fig. 7. Thus, overall, we replicated the results of Martin and Raisamo in the first block, but the additional second block indicated that with training the task completion time with the mouse configuration caught up with the Trackmouse configuration.

6.5.2. Error rate

Errors were counted on per-task basis. That is, if at the end of the task there were red dots outside the ellipse or black dots within the ellipse, the task was counted as an error. Overall there were fewer errors in the second block (3.9%) than in the first block (6.3%), ($F_{1,11} = 2.3$, $p < 0.01$), but no other statistically significant differences.

6.5.3. Degree of parallelism

We measured the degree to which the two cursors were operated simultaneously. For this, we divided the task completion time into 100 ms periods and counted the number of these periods in four categories: (1) first cursor movement, (2) second cursor movement, (3) movement of both cursors, and (4) no cursor movement. We consider the third category where both cursors moved within 100 ms of each other to represent parallel use of both input streams. Both cursors were used over 20% of the task completion time with the two-handed interface and with the Trackmouse. Our analysis method does not reveal whether the simultaneous movement was intentional or accidental.

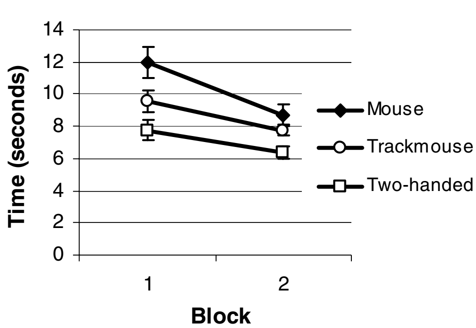

Fig. 7. Task completion time in the ellipse task.

6.5.4. Subjective ratings

Generally, the two-handed condition received more positive responses than the mouse and the Trackmouse ($p < 0.05$ in all cases). There were no significant differences in the ratings between the mouse and the Trackmouse. An exception was the question that asked the participants to rate how comfortable the force required for operating the device was. In this case the Trackmouse was rated lower than the mouse and the two-handed interface (in both cases $p < 0.01$). When asked for a reason for this assessment, the participants reported that the ball in the Trackmouse was uncomfortably sensitive.

7. Toolbar task

Experiment 2 showed that in a combined and asymmetric dual-stream task, some initial improvement in performance was achieved by using a Trackmouse rather than a conventional mouse. Unfortunately, the advantage of using a Trackmouse in the task of experiment 2 vanishes with training. Therefore, if there are tasks where the Trackmouse can outperform the mouse in the long term, they are probably of different kind.

We chose to study the repetitive use of toolbars. It occurs often in computer use, for example, in precise pixel manipulations in graphics software and when changing the formatting of a text document. This kind of a task can be considered symmetric, although when completed with two cursors, one for selecting the tools, and one for working in the working area of the display, the accuracy requirements for the two cursors may not be the same. However the low-level task, i.e. pointing, is exactly the same for both cursors. There is no coarse scale positioning and setting the context by one and working in the context by the other. Both cursors point and click relatively independently at targets on the display.

Moving between the toolbar and the working area lends itself well to modeling with Fitts' law (Fitts, 1954; MacKenzie, 1992; Soukoreff and MacKenzie, 2004). To explain our rationale for experimenting with this task we will now compare the efficiency of the motor component of doing the toolbar task with one cursor and with two cursors. The motor component is only a part of pointing behavior. Especially in the beginning when users are learning to use a new pointing device, cognitive processing may occupy significant portion of the task completion time. However, we begin with modeling the motor component and discuss the real performance that includes time for cognitive operations later in the context of experiments 3 and 4.

Fig. 8 shows pointer movements in a user interface of software such as a painting program. There is a toolbar with six tools (one of them highlighted in gray) and two drawing objects. The arrows in the top half of Fig. 8 show cursor motion in one cycle of the object-toolbar-object motion that we are interested in. The bottom half of Fig. 8 shows cursor motion in the same cycle when two cursors are used.

Using Fitts' law we can calculate the index of difficulty (ID) for the pointing cycle using the formula

$$\mathrm{ID} = \log_2\left(\frac{A}{W} + 1\right)$$

where A is the distance from the starting point to the target and W is the width of the target. For the sake of simplicity we are ignoring details regarding the angle of approach (Accot and Zhai, 2003). In essence, this means resorting to the smaller-of model of MacKenzie and Buxton (1992) where the smaller of target height and width is used as W. Throughout this section we assume a target size of 30 units.

With short distances ($A = 200$) moving the cursor from the working area to a tool has ID equal to $\log_2(200/30 + 1) \approx 2.9$ bits and moving back to the next object in the working area has the same ID. The sum of these is 5.9 bits. Doing the same calculations for a longer distance ($A = 600$) between the toolbar and the drawing objects we get 8.8 bits.

As seen in the lower half of Fig. 8 the distances covered by the cursors are much smaller if two cursors are used for the same task and they are conveniently placed in the beginning. Doing the Fitts' law computations ($\log_2(30/30 + 1) + \log_2(60/30 + 1)$), we get 2.6 bits regardless of the distance between the toolbar and the drawing objects.

In comparison to the one-pointer case, using two pointers seems to save 3.3 bits in the short distance case and 6.2 bits in the long distance case. If we assume a bandwidth of 5 bps for the pointing devices being used, these numbers can be converted into time using the formula $t = \mathrm{ID}/\mathrm{IP}$, where t is the time, ID is the index of difficulty, and IP is the index of performance (bandwidth). The result is that we save 0.7 s with the shorter distance and 1.2 s with the longer distance.

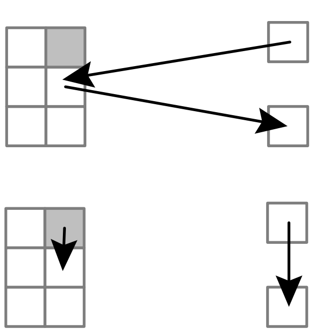

Fig. 8. Cursor motion in one cursor (top) and two cursor (bottom) toolbar usage.

Switching attention between the different pointers and input streams takes time and two input streams probably cannot be operated at the same 5 bps rate using only one hand. Unfortunately, we did not know how to quantitatively model attention switching and one-handed two-pointer use. This is why we needed to conduct experiment 3 to measure user performance in the toolbar task.

8. Experiment 3

Our model of the toolbar task ignores the cognitive and mechanical difficulties in operating two cursors. The cost of these difficulties in operating the two cursors with the Trackmouse and the two-handed interface may be greater than the motor benefits predicted by the model. The aim of this experiment was to measure which factors dominate this trade-off for users without much experience in trackball-mouse use or two-handed use of a trackball and a mouse.

8.1. Participants

The participants from experiment 2 participated also in this experiment except for one who was ill. She was replaced by one of our pilot test participants who had roughly the same amount of experience with the devices being used as well as same sex and roughly the same age.

8.2. Apparatus

The computer, the input devices, and the physical surroundings were the same as in experiment 2. The software was different, but it performed the same functions. That is, it presented the task and saved user input for later analysis.

8.3. Task

A trial started when it was displayed and ended when the participant clicked a mouse button outside the toolbar area. The first trial was displayed when the software was started, and the following trials in a block were displayed as soon as the previous trial ended. The trial ended upon a click regardless of whether the participant hit the target or not.

For each trial the participant had to observe the character displayed in the target, click on the corresponding toolbar item, and click on the target. The targets were 20-pixel squares and the toolbar buttons were 30-pixel squares. In the mouse configuration all clicking was done with the left button of the Trackmouse using only one cursor. In the Trackmouse and two-handed configurations two cursors were available. The first one worked just as in the mouse configuration. However, it was not to be used for toolbar selections. Instead, the ball, operated by the left hand in the two-handed configuration or by the thumb of the right hand in the Trackmouse condition was used for moving the second cursor and the right button on the Trackmouse to click. The second cursor was blue so that the cursors would not be confused easily. The blue cursor only moved within the toolbar strip that occupied the left edge of the display (as shown in Fig. 9).

The "Near", "Middle", and "Far" texts and boxes around them in Fig. 9 were not shown to the participants. They illustrate the areas where the targets appeared in the different distance conditions. In Fig. 9 the target is above the "Far" box and to the right from the "Middle" box. This screenshot is from the "Everywhere" block where targets were placed randomly over the whole screen outside the toolbar area. "Everywhere" blocks were included as the first and last blocks in the experiment.

8.4. Procedure

The participants performed 5 blocks of 30 trials with each of the device configurations totaling $5 \times 3 \times 30 = 450$ trials per participant. There were no breaks between the trials within a block. The participants could take a break in the middle of a block by pressing the 'P' key on the keyboard. This option was used only once and by only one participant. The trial, during which the pause occurred, was excluded from the data analysis. There were short breaks between the blocks while the experimenter re-started the software with the new configuration.

Timing started when a new trial was displayed and ended when the trial was completed by clicking the left Trackmouse button. A new trial was displayed instantly when the previous trial was completed. The first trial in a block was excluded from data analysis because its timing was abnormal due to the time that the participant needed for grasping the device and getting started.

The differences between the five blocks with each device were in the location of the targets. In the first block and in the last block with each device the targets were placed randomly in the whole working area. In the second, third, and fourth blocks the targets were placed randomly within one of the 256×256 pixel areas shown in Fig. 9. The order of the Near, Middle, and Far blocks was balanced between participants. The aim was to find the crossover point in distance where use of the dual-stream configurations would start to pay off. We also wanted to measure the learning rate with different device configurations to see if the learning trends in this task seemed more promising for Trackmouse than they did in experiment 2.

8.5. Results

The results from blocks 2–4 will be described first followed by the results from blocks 1 and 5. The separate analysis was necessary because of the different treatment of learning effects. The order of blocks 2–4 was balanced to get learning-neutral data. In blocks 1 and 5 the goal was to measure the effect of learning.

8.5.1. Task completion time

Fig. 10 shows the task completion times for blocks 2–4. The task completion time for the mouse follows a logarithmic curve as suggested by Fitts' law. The user performance with the Trackmouse and the two-handed configuration were not affected by the distance as much as the mouse configuration, as suggested by the model above.

A 3×3 (distance × device configuration) repeated measures ANOVA shows significant effects of device

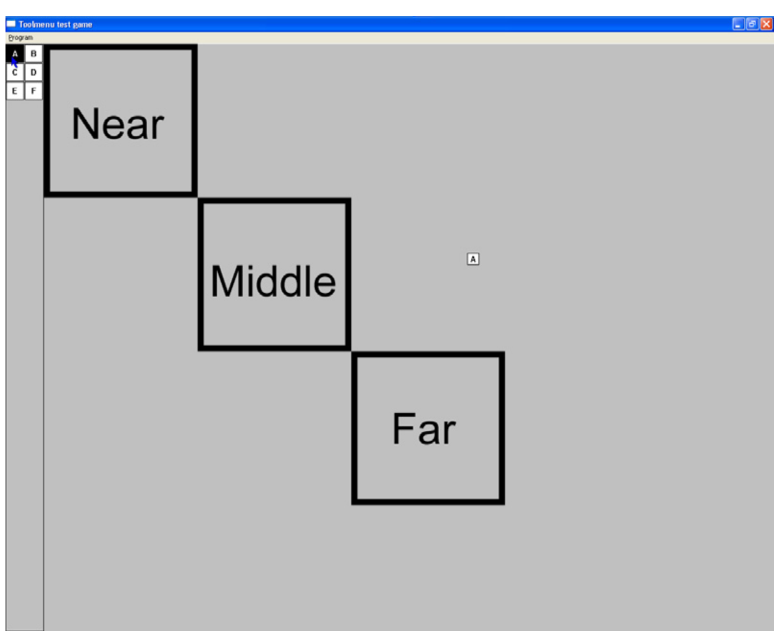

Fig. 9. The toolbar task and three areas where targets appeared.

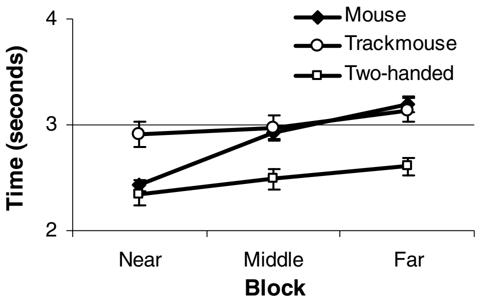

Fig. 10. Task completion time in the toolbar task.

configuration ($F_{2,22} = 44.7$, $p < 0.01$) and distance ($F_{2,22} = 23.3$, $p < 0.01$) and a significant interaction of these ($F_{4,44} = 16.4$, $p < 0.01$).

Based on Fig. 10 it is clear that using the mouse configuration was faster than using the Trackmouse configuration in the short distance block ($t_{11} = 6.1$, $p < 0.01$). At long distances the mean task completion time for the Trackmouse configuration (3.14 s) was slightly lower than for the mouse configuration (3.19 s) but the difference was not statistically significant ($t_{11} = 0.5$, $p = 0.6$). At short distances the two-handed configuration was slightly faster than the mouse configuration, but the difference is not statistically significant ($t_{11} = 1.4$, $p = 0.2$). At long distances the two-handed configuration was statistically significantly faster ($t_{11} = 8.5$, $p < 0.01$) than the mouse configuration.

Fig. 11 shows the task completion time for Blocks 1 and 5. A 2×3 (block × device configuration) repeated measures ANOVA indicates an effect of block ($F_{1,11} = 311$, $p < 0.001$), an effect of device configuration ($F_{2,22} = 42.1$, $p < 0.001$), and an interaction of block and device configuration ($F_{2,22} = 22.7$, $p < 0.001$).

In Fig. 11 we can see what causes the interaction effect. The training improved the task completion time with the mouse only slightly. The task completion time with the Trackmouse configuration improved a lot, and the improvement with the two-handed configuration was in between these two extremes. Paired samples *t*-tests show that in block 1 Trackmouse configuration was slower than the mouse configuration ($t_{11} = 6.8$, $p < 0.001$) and the two-handed configuration ($t_{11} = 6.7$, $p < 0.001$), but there was no statistically significant difference between the two-handed configuration and the mouse configuration ($t_{11} = 0.6$, $p = 0.6$). In block 5 the two-handed configuration was faster than the mouse configuration ($t_{11} = 3.4$, $p < 0.01$), the mouse configuration was faster than the Trackmouse configuration ($t_{11} = 2.9$, $p < 0.05$), and the two-handed configuration was also faster than the Trackmouse configuration ($t_{11} = 4.7$, $p < 0.001$). An interesting question regarding the Trackmouse configuration is whether performance with it would have continued to improve faster than the performance with the mouse configuration if the training had continued. Experiment 4 was conducted to answer this question.

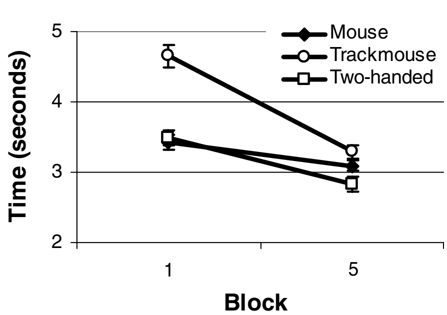

Fig. 11. Task completion time in the toolbar task.

8.5.2. Error rate

An error was recorded whenever the target was missed, or hit with the wrong toolbar button selected. The average error rate was 3%. We found no statistically significant effects of device configuration or distance.

8.5.3. Degree of parallelism

Simultaneous use of the two pointers took place, but to a lesser extent than in experiment 2. The percentage of 100 ms windows with both cursors moving was in the range of 15–21% for the two-handed configuration and in the range of 8–14% for the Trackmouse configuration. Experiment 2 was synergistic and experiment 3 was alternate. Thus, the results for the degree of parallel use of the input streams agree with what could be expected from tasks in these classes.

8.5.4. Subjective ratings

The subjective evaluation form of experiment 2 was used also for experiment 3. The results were similar; the two-handed interface was preferred over the others with no difference between the mouse and the Trackmouse. The exceptions to this rule were the question on finger fatigue where there was no difference between the two-handed configuration and the mouse ($z = -1$, $p = 0.3$) and the question on accuracy where the mouse was rated higher than the Trackmouse ($z = -2.3$, $p < 0.05$).

9. Experiment 4

The results of experiment 3 are not what one would expect based on the preceding modeling. The two-handed interaction did deliver some of the predicted performance advantage, but the Trackmouse did not. Our explanation for this was that two-handed use of two pointers does not require as much of learning of new motor skills as Trackmouse operation does. People are used to holding objects in both hands and manipulating things with them and that is why the two-handed configuration is faster from the beginning.

To see whether learning in the Trackmouse configuration continues to be faster than in the mouse configuration over a longer period of time, we conducted a longitudinal experiment.

9.1. Participants

Six (male) participants were recruited from the staff of our universities. The average age of the participants was 34 years (range 24–48). All were experienced mouse users with no or only cursory experience with trackball-mice.

9.2. Apparatus

To lessen the burden on the participants the experiment was conducted in their offices or other convenient locations. Thus, the physical surroundings varied. However, the surface under the Trackmouse was standardized by taping a large (A3) sheet of white paper on the table. Also, the experiment was always conducted using the same computer (Acer TravelMate C312 laptop) and an external 17-inch LCD display in 1280×1024 pixel resolution.

9.3. Procedure

The task consisted of blocks of 50 toolbar-target selection pairs. Eight blocks (four with one cursor and four with two cursors) were completed in pre-test to measure the starting level of the participants. The test blocks were labeled "everywhere", "near", "middle", and "far". In the everywhere-block the targets appeared in a 768×768 pixel area. In the other blocks the targets appeared in 256×256 pixel areas shown in Fig. 9. The everywhere-block was always performed first to emulate the procedure of experiment 3 where participants had one block of training before the distance controlled blocks. The order of the near, middle, and far blocks was balanced between subjects. A post-training test with the same block and device configuration order was done at the end of the experiment to measure user performance under the same conditions after training.

Between the pre-test and the post-test the participants performed four sessions of training. The training sessions consisted of 10 blocks of training using the Trackmouse as a mouse only, and 10 blocks of training with two cursors in the Trackmouse mode. The middle-distance condition (see Fig. 9) was used for the training. The order of mouse and Trackmouse was balanced between participants so that half of them began with the ten blocks using the mouse configuration and the other half began with ten blocks using the Trackmouse configuration. The order of the device configurations was also balanced within subject so that it was reversed on consecutive sessions.

Thus, per device configuration the participants had one block (i.e. 50 targets) of training before the first distance controlled pre-test block. Before the first training block they had four blocks (i.e. $4 \times 50 \times 200$ targets) of training, and before the first post-test block they had 44 blocks of training ($44 * 50 = 2200$ targets).

At the end of the last session the participants were asked to rate their impressions on mouse and Trackmouse usage experience using the same form that was used in experiment 3. The first session lasted about 90 min, the second and third about 45 min, and the last about 80 min.

One of the benefits of the Trackmouse prototype that we used in this experiment was that its shape allowed a fairly natural and efficient grip as shown in Fig. 12. If the user places his or her hand so that the middle and ring finger are used to press the buttons, the forefinger is left on the left side of the device and can be used to push the device rightwards when using the mouse functionality. This is important because the thumb should always be ready to operate the ball, and not be used for anything else. We instructed the participants to use this grip when using the Trackmouse configuration. They were free to use whatever grip they wished when using the mouse configuration.

Fig. 12. An efficient grip for the TrackMan Trackmouse prototype.

To ensure meaningful results, we controlled the error rate during the training. This happened by monitoring the progress of the participants and by instructing them when necessary at the beginning of training sessions 2, 3, and 4. We computed the results of the previous session and discussed them with the participants before the session. If participants favored accuracy too much (i.e. did not make errors in some blocks), we instructed them to work faster in the following training session. Similarly, when they made too many errors (i.e., more than five per block) we instructed them to be more accurate. This procedure helped to maintain the same error rate with both device conditions. This means that the task completion times can be compared and the faster configuration is the more efficient one. Additionally, as a result of our interventions, the participants were likely to be working at the limit of their capabilities. They were working fast enough not to hit the targets perfectly all the time, but not so fast that their performance would be out of control.

9.4. Results

9.4.1. Pre- and post-test data

Fig. 13 shows the task completion times per target for the distance controlled block in the pre-test and in the post-test. In the pre-test the Trackmouse configuration was slower than the mouse configuration. In the post-test there either was no statistically significant difference or the Trackmouse was faster.

A 3 × 2 (distance × device configuration) repeated measures ANOVA was performed for pre- and post-test data separately. In the pre-test data all factors had a significant effect. The Trackmouse configuration was slower than the mouse configuration ($F_{1,5} = 7.9$, $p < 0.05$), increasing distance between the toolbar and the target increased task completion time ($F_{2,10} = 7.4$, $p < 0.05$), and there was a significant interaction of distance and device configuration ($F_{2,10} = 10.3$, $p < 0.01$). The interpretation of the interaction is as in experiment 3. Increasing distance has a smaller effect on task completion time in the Trackmouse configuration than in the mouse configuration.

In the post-test results the effect of distance was again significant ($F_{2,10} = 123.7$, $p < 0.001$). The effect of the device configuration was not ($F_{1,5} = 1.2$, $p = 0.3$), but there was a significant interaction of distance and device configuration ($F_{2,10} = 20.8$, $p < 0.01$). With the help of Fig. 13, we can interpret the interaction again as smaller effect of distance on the Trackmouse configuration. Pair-wise comparisons within the distance conditions show that with the near and middle distances the difference between the device configurations was non-significant ($t_5 = -0.9$, $p = 0.4$, and $t_5 = 1.3$, $p = 0.3$), but with long distance the Trackmouse configuration was faster ($t_5 = 4.2$, $p < 0.01$).

Fig. 14 shows the error rates in the pre-test and post-test. The 3 × 2 (distance × device configuration) ANOVAs on the pre- and post-test error data revealed no statistically significant effects. Based on Fig. 14 it appears that the participants maintained roughly same error rate in both device configurations.

The results of the "everywhere" block are not shown in Figs. 13 and 14. The effect of the training, however, was similar. Initially the mouse was faster (2.6 s vs. 3.4 s, $t_5 = 5.6$, $p < 0.01$), but after the training the difference was no longer statistically significant (2.11 s vs. 2.16 s, $t_5 = 0.7$, $p = 0.5$). There were no statistically significant differences in error rates.

9.4.2. Longitudinal data

The development of the task completion time and error rate over the training sessions is shown in Figs. 15 and 16. Using the Trackmouse configuration was initially slower. The participants got faster in both device configurations, but the speed with the Trackmouse configuration devel-

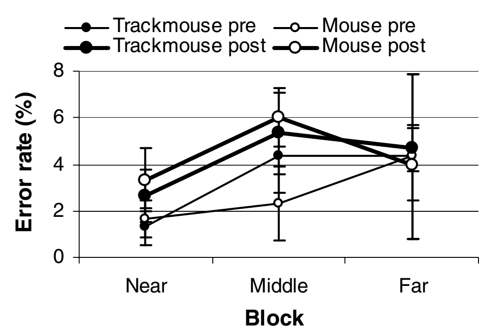

Fig. 14. Error rates in pre- and post-test.

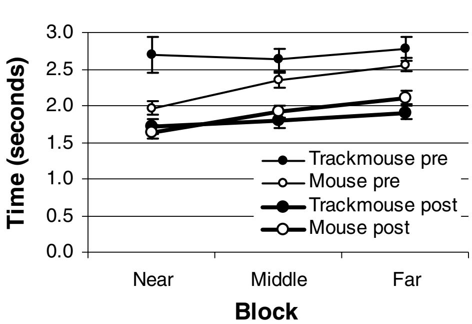

Fig. 13. Task completion time per target in pre- and post-test.

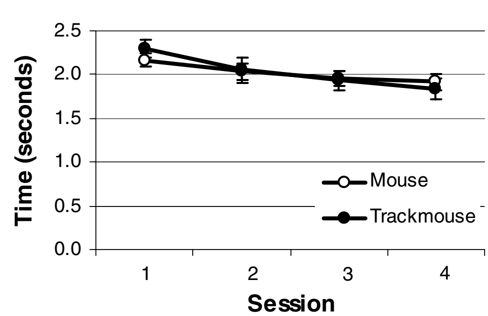

Fig. 15. Development of average task completion time over the training blocks.

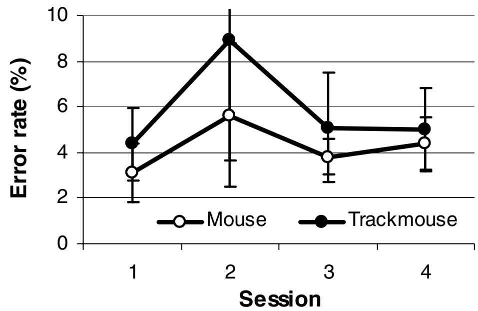

Fig. 16. Development of average error rate over the training blocks.

oped faster throughout the training. A 4 × 2 (session × device configuration) ANOVA on the data shown in Fig. 15 supports this interpretation. The effect of device configuration was not significant ($F_{1,5} = 0.02$, $p = 0.2$), but both the effect of the session ($F_{2,10} = 29.0$, $p < 0.01$) and the interaction of device configuration and session ($F_{3,15} = 5.8$, $p < 0.05$) were significant. To further investigate the interaction, we performed pair-wise *t*-tests for the difference between the device configurations within each session. The tests showed no statistically significant differences. This result agrees with the post-test result where the difference in the middle distance block was not statistically significant.

The error rate for the Trackmouse configuration seems higher in Fig. 16, but due to the high variability of the error rate data a 4 × 2 (session × device configuration) repeated-measures ANOVA showed no statistically significant effects of session or device configuration or their interaction.

9.4.3. User impressions and strategy

The questionnaire given at the end of the experiment asked the participants to rate their experience during the last session only. The data revealed only one statistically significant difference between the device configurations. This difference was in question 3 that asked for their opinion on the "overall operation of the input device" on a seven step scale from "very difficult" (1) to "very easy" (7). The mouse configuration was rated to be easier to operate ($M = 6.7$) than the Trackmouse configuration ($M = 4.7$), ($z = 2.2$, $p < 0.05$). In other words, the participants still found the Trackmouse configuration more difficult to use than the mouse configuration after about two hours of training.

Based on our own experience and by observing the participants we found that there are two strategies for using the thumb on the ball. One is to lift the thumb when using the mouse and the other is to keep the thumb on the ball at all times. The latter seems more efficient. However, in the beginning lifting the thumb when using the mouse functionality feels attractive because some unintended ball movements can be avoided by doing so. With training, however, the unintended ball movements seem to be under better control, or possibly recovering from them is quicker. Overall, it appears very difficult to surpass mouse efficiency with the thumb-lift strategy. Starting from participant two, we explicitly instructed the participants to keep the thumb on the ball at all times except when correcting the overall position of the ball.

Another effect mentioned by some participants is that when the cursors were close to each other, the task felt easier. Balakrishnan and Hinckley (2000) reported that performance in symmetric bimanual work (with two cursors) is more efficient when both cursors can be seen accurately at the same time. It appears that this finding holds for one-handed symmetric two-stream manipulation as well, at least on the level of user impressions. We do not have quantitative data to support this finding except for the hint offered by the dip of the error rate curves in the "near" condition in Fig. 14. However, if real, the added efficiency at short distances due to visual integration of the two cursors helps to compensate for the relative weakness of two-cursor use at short distances. In other words, when the distance between the cursors is long, the saved motion makes Trackmouse usage efficient, and when the distance is short, the ability to track the two cursors visually may help to catch up to the same level of performance with conventional single-cursor mouse use.

10. Summary and discussion

10.1. Experiment 1

Our first experiment compared two Trackmouse prototypes: one with a thumb-operated ball, and one with a finger-operated ball. The results show no difference in pointing performance between the prototypes in combined use of the mouse and the trackball or in trackball use. The prototype with the thumb-operated ball was superior when only the mouse functionality was used. This may be because of its mouse-like shape and button placement. If this is the explanation, it is also a warning against broad generalizations. It may be that some of our results are specific to the particular devices that we used. Experiments with a broader spectrum of different trackball-mouse body shapes are needed in further work.

The results also suggest that in one-handed tasks Fitts' law describes user performance rather well ($R^2 \approx 0.9$) even if the task is completed with two pointers.

Regardless of the reasons for why the prototype with the thumb-operated ball was better, its use in the other experiments was justified. We saw no reason for using other than the best prototype known to us.

10.2. Experiment 2

The results for the first block of tasks confirmed the findings of Martin and Raisamo (2004). By using a Trackmouse the users initially performed better than by using only the mouse functionality in a surround-with-ellipse task. However, the second block revealed new information.

In the original experiment by Martin and Raisamo the task was repeated 15 times. We ran two 25-task blocks. In the second block the difference in task completion times between the Trackmouse mode and the mouse mode was no longer statistically significant.

This may suggest that with increasing training the advantage measured for Trackmouse by Martin and Raisamo and block 1 in experiment 2 diminishes. Extrapolating the trends seen in Fig. 7 reveals that with continued training the Trackmouse configuration might turn out to be slower than the mouse configuration. A further longitudinal study on expert performance in this task is needed to find out whether the trends continue in this way with further training.

The task in experiment 2 was similar to the task used by Leganchuk et al. (1998) in the respect that the main advantage of the two-handed and Trackmouse configurations over the mouse configuration was the reduced need for mode switching. In the task by Leganchuk et al. the participants needed to switch the control point by actually moving the cursor and pressing the mouse button when the cursor was over the control point. In experiment 2 a button press immediately switched the active control point. This diminished the difference in motor activity between the mouse configuration and the other configurations. However, the difference in cognitive work remained. In the mouse configuration the participants had to decide which control point would best serve their goals, and then switch accordingly. Note, however, that the control points were different. Leganchuck et al. used opposite corners of the bounding box, whereas we used the center and the upper left corner. Also, our target was a group of dots, whereas Leganchuk et al. used geometric shapes. Also, the bounding box was not directly involved in experiment 2. It was never shown, and the participants most likely felt like they were interacting directly with the ellipse rather than with the bounding box. Therefore, our results apart from the Trackmouse configuration are not a replication of the work by Leganchuck et al., although similar.

10.3. Experiment 3

In the toolbar-task of experiment 3 the two-handed configuration was the most efficient in the long distance block. In a similar experiment Hinckley et al. (2002) did not observe this. This is probably because in their study the menu could be repositioned close to the objects. Their task was similar to our "near" block where we did not find a statistically significant difference between the mouse and the two-handed configuration.

In experiment 3 the mouse configuration and the Trackmouse configuration exhibited the expected relationship. Namely, increasing distance between the toolbar and the targets hurts user performance with the mouse configuration more than the performance with the Trackmouse configuration. However, the participants, who were experienced mouse users, performed better with the mouse configuration than with the Trackmouse configuration with which they did not have experience.

Overall, for the comparison of the mouse configuration and the Trackmouse configuration, experiment 3 showed that when there was a statistically significant difference, it was in favor of the mouse configuration. However, the data in Fig. 11 suggest that learning with the Trackmouse configuration may be faster than with the other configurations. We observed a similar trend in experiment 1. Experiment 4 was conducted to see if this trend continues beyond the very short training in experiments 1 and 3.

10.4. Experiment 4

Experiment 4 was a repetition of the toolbar task of experiment 3 with about two hours of training per device configuration. The goal was to see whether users' performance with the Trackmouse configuration exceeds their performance with the mouse configuration as they begin to develop a device-specific skill in Trackmouse operation. The results show that with training, the performance with the Trackmouse configuration does indeed improve more than the performance with the mouse configuration. However, in the end the Trackmouse configuration outperformed the mouse only when the distance between the toolbar and the targets was long.

With further practice users are likely to continue improving their performance with both device configurations. Fig. 17 shows the data of Fig. 15 extrapolated with the best fitting log-log linear regression curves. Assuming that learning proceeds in the future as it has so far, we can estimate the effect of further training as shown in Fig. 17. Naturally, we should not trust these extrapolations very far from the measured data.

In Fig. 13 it seems that when the distance between the toolbar and the working area is longer than about 300 pixels, using the Trackmouse configuration is more efficient. Even if performance with the Trackmouse configuration improves slightly with further training, it appears that in this task the two-cursor Trackmouse configuration will never be more efficient than one-cursor mouse use on short distances. Whether using the Trackmouse configuration in this task is a good idea depends on the typical distance between the toolbar and the working area. Sometimes it

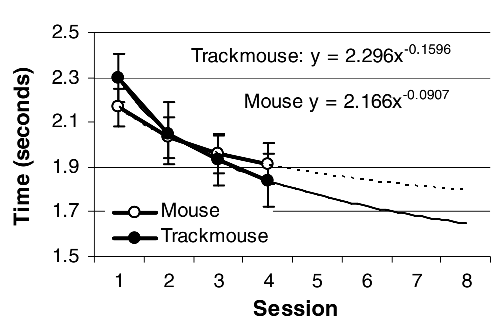

Fig. 17. Extrapolations of the learning curves in Fig. 15.

is very short, but at other times, for example, when using multiple monitors, it may be very long.

10.5. General discussion

We found conflicting results on the effect of training on the relative efficiency of two-stream trackmouse configuration and the same device used as a mouse only. In experiment 2 the Trackmouse was initially more efficient, but with training the mouse configuration seemed to improve more than the Trackmouse configuration. In experiments 1, 3, and 4 the Trackmouse configuration seemed to benefit more from training. However, the only circumstance in these three experiments where the Trackmouse configuration actually outperformed the mouse was the long distance condition at the end of experiment 4.

The effect of training on the relative efficiency of the Trackmouse seems to depend on the task. In this respect Trackmouse is probably similar to other multiple-degrees of freedom pointing device configurations. The match of pointing devices and the actions performed with them is sensitive to design. For example, counter-intuitive or changing stimulus-response compatibility can destroy the potential benefits as demonstrated, for example, by Latulipe et al. (2005a,b, 2006).

When measured, the user performance with two-handed interaction techniques was remarkable in comparison to the other configurations. The participants were not experienced users of two pointing devices, but still in the two-handed configurations they outperformed themselves in the one-handed mode – sometimes by a large margin. This, once again, makes one ask why two-handed use of pointing devices is not the norm in pointing-intensive application areas such as graphics. Maybe this is because most users do not have two pointing devices to use and software developers do not want to put in the effort of supporting only a few enthusiasts. Or maybe the second hand is better used operating the keyboard. One possibility for increasing the market for software that supports two-handed use of pointing devices is that one-handed two-stream devices work as a bridge that helps to build the critical mass of users of two-stream pointing. If some of the benefits of two-handed use can be achieved with one-handed use of two pointers, some users that would not invest in two pointing devices may acquire a one-handed two-stream device. If enough users do this, there may be enough return for investment so that two-handed interfaces finally become widely available.

Comparing the results of experiments 3 and 4, we notice that training improves user performance even with the mouse that the participants were very experienced with. The average time per toolbar-target cycle in the middle-distance task improved from about 2.3 s to 1.9 by the end of experiment 4. This is a sizable improvement in a task that the participants should know well having pointed at targets with the mouse for several years. This improvement is about the same magnitude as the advantage that we measured for the two-handed conditions in experiments 2 and 3. If something is to be concluded of this, it is that we should avoid generalizing results from experiments that measure only novice or only expert performance and we should not assume that experience with a device implies excellent skill in all possible tasks with it. There are task-specific strategies and skills that once learned improve the performance of even an experienced user.

In the light of recent results, dual-stream input, whether used by two hands or one, seems to show promise even when not strictly adhering to the asymmetric guidelines based on Guiard's kinematic chain model. For example, symmetric use of two cursors can outperform traditional mouse use in generic graphical user interface usage as well as in more exotic tasks explored by Latulipe et al. (2005a,b, 2006). Therefore, the kinematic chain model is probably best understood as a descriptive model of many tasks and as a useful guideline for designing interaction techniques, but it should not be applied rigidly to limit innovation in user interfaces. Sometimes other considerations such as saved pointer movement (as demonstrated above) or tasks that match the input devices particularly well (Latulipe et al., 2006) can result in useful designs involving symmetric use of input devices.

10.6. Limitations of the experiments

We have mostly measured novice performance with dual-stream input configurations and compared it to performance with the mouse that the participants were familiar with. Our final experiment measured the development of Trackmouse usage performance over about two hours of use. However, real-world users that have a trackball-mouse as a primary pointing device will quickly exceed the two-hour period by multiple orders of magnitude. Therefore, expert performance with trackball-mice can later develop in ways that were not possible to study within the scope of this work.

While our experiments did give results on user performance with trackball-mice, they could not reveal much on other aspects of the devices that are equally important in real usage situations. The users in the experiments were volunteers working in research units within computer science departments. They may have had a positive attitude towards such experimentation and possibly more interest in new pointing devices than the general population. This may be reflected in the subjective data that we collected. We also did not test usage where tasks that favor trackball-mice are interspersed with other kinds of tasks that may not even involve pointing. Consequently, we do not know how easy it is for the users to integrate efficient trackball-mouse operation in their daily desktop computing behavior. This is left for future work.

We always compared mouse and Trackmouse performance using the same device. In mouse use the ball on the device was just ignored. This allowed us to compare user performance in different usage modes rather than with different devices. However, it is possible that our Trackmouse prototypes were not very good in the mouse mode. We did take steps to make it unlikely that this was the case. We tested two prototypes and rejected the one with the worse mouse performance from further experiments. We also measured the throughput of the prototypes in mouse mode. The values recorded in experiment 1 (5.4 and 4.7 bps) are in the range that one usually finds for normal mice in the ISO multi-directional task. In their review Soukoreff and MacKenzie (2004) report two studies that used the same task with mice. Both produced a result of 4.9 bps. Based on this, it appears that our prototypes performed well when used in the mouse only mode.

10.7. Future work

It should be noted that we, like some other researchers (Balakrishnan et al., 1997; Latulipe et al., 2006), have not included the combined use of a keyboard and a pointing device in our experiments. As noted earlier in the context of Sutherland's Sketchpad, keyboard commands and mode-switching combined with a pointer is an old, effective, and popular way of accomplishing complicated tasks involving pointing. Omitting this device combination in our experiments was intentional. We wanted to first establish whether a trackball-mouse use can be more efficient than conventional mouse use. Now that we know of two conditions where the Trackmouse configuration outperforms the mouse (beginning of experiment 1, and long distance condition in the end of experiment 4), we can proceed to compare the Trackmouse to a keyboard and mouse combination under the same conditions.

Comparing trackball-mouse to the keyboard-mouse combination involves the same difficulty as the comparison to the conventional mouse use. All available users are experienced in the use of the keyboard-mouse combination and need a lot of training in trackball-mouse use before there is hope of demonstrating advantages of trackball-mouse use.

Multiple degrees of freedom input devices are often used in gaming. The four degrees of freedom on a trackball-mouse match, for example, the number of degrees of freedom needed in moving and aiming in typical first person shooter games. We have begun work to experiment with different mappings of these degrees of freedom and to evaluate trackball-mice as game controllers. Gaming itself is a significant area of user interface industry. We also see games as an educational environment and a way to introduce users of "serious" software to different input devices and the benefits of multi-degree of freedom input and multimodal user interfaces.

11. Conclusions

Users seem to be able to outperform a traditional mouse with a trackball-mouse only in some tasks and only under specific conditions. Novices performed well in the beginning of an ellipse moving and resizing experiment, and experienced users did well in a long-distance condition at the end of a symmetric two-cursor toolbar experiment. When tested, two-handed use of a mouse and a trackball outperformed the mouse and the Trackmouse.

It appears that there are many things in graphical user interfaces that can be done more efficiently with two pointing devices. We agree with Latulipe et al. (2006) on that adding support for a second pointing device would be a good idea. In this context the work on trackball-mice is significant. If user interfaces were modified so that they support or even require the use of two pointing devices a trackball-mouse can be useful in situations where two-handed use of pointing devices is not possible. Such situations may arise, for example, because of tasks that require frequent use of keyboard, or because of desk-space constraints and in semi-mobile situations with laptop computers.

Based on experiment 1 it seems that the shape of the device body makes a difference in trackball-mouse performance. In this experiment only the performance of the Marble Mouse device when used as a mouse seemed to suffer, but in other tasks similar effects may emerge. For example if the task has a strong directional bias, the placement of the ball and the button(s) associated with it may make some configurations easier to operate than others through better stimulus-response compatibility.

There are technical obstacles to be tackled before trackball-mice can be useful. General multiple cursor support in GUI libraries is needed as well as software that utilizes the four degrees of freedom input in other ways. It remains to be seen whether these conditions will be satisfied and if so, whether trackball-mice become main stream input devices. Our results show that if this were to happen, the pointing performance of the users would probably not suffer in the long run.

Returning to our original motivation, we now have data on user performance with a trackball-mouse. The data show that trackball-mice are not significantly more efficient than conventional mice. However, they are not significantly worse either. The main obstacle in effective trackball-mouse usage is not user performance with such devices, but the rarity of systems that support useful ways of using them. The history of human-computer interaction research is full of user interface ideas that seem buried in libraries and never see the light of day due to unfavorable circumstances. On the other hand, every now and then some of them are found useful when the mainstream computing paradigms shift to a fruitful direction. We look forward to a shift towards increasing use of multiple pointers and pointing devices.

References

Accot, J., Zhai, S., 2003. Refining Fitts' law models for bivariate pointing. In: Proc. CHI 2003, ACM Press, pp. 193–200.

Annett, J., Annett, M., Hudson, P.T.W., Turner, A., 1979. The control of movement in the preferred and non-preferred hands. Quarterly Journal of Experimental Psychology 31, 641–652.

Balakrishnan, R., Baudel, T., Kurtenbach, G., Fitzmaurice, G., 1997. The Rockin'Mouse: integral 3D manipulation on a plane. In: Proc. CHI '97, ACM Press, pp. 311–318.

Balakrishnan, R., Hinckley, K., 2000. Symmetric bimanual interaction. In: Proc. CHI 2000, ACM Press, pp. 33–40.

Balakrishnan, R., Patel, P. 1998. The PadMouse: facilitating selection and spatial positioning for the non-dominant hand. In: Proc CHI '98, ACM Press, 1998, pp. 9–16.

Buxton, W., Myers, B.A., 1986. A study in two-handed input. In: Proc. CHI '86, ACM Press, pp. 321–326.

Casalta, D., Guiard, Y., Beaudouin-Lafon, M., 1996. Evaluating two-handed input techniques: rectangle editing and navigation. in: Ext. Abstracts CHI '99, ACM Press, pp. 236–237.

Chatty, S., 1994. Extending a graphical toolkit for two-handed interaction. In: Proc. UIST '94, ACM Press, pp. 195–204.

Dillon, R.F., Edey, J.D., Tombaugh, J.W., 1990. Measuring the true cost of command selection: techniques and results. In: Proc. CHI '90, ACM Press, pp. 19–25.

Douglas, S.A., Kirkpatrick, A.E., MacKenzie, I.S., 1999. Testing pointing device performance and user assessment with the ISO9241, Part 9 standard. In: Proc. CHI '99, ACM Press, pp. 215–222.

Fitts, P.M., 1954. The information capacity of the human motor system in controlling the amplitude of movement. Journal of Experimental Psychology 47, 381–391.

Flowers, K., 1975. Handedness and controlled movement. British Journal of Psychology 66, 39–52.

Guiard, Y., 1987. Asymmetric division of labor in human skilled bimanual action: the kinematic chain as a model. The Journal of Motor Behavior 19, 486–517.

Hinckley, K., Czerwinski, M., Sinclair, M., 1998. Interaction and modeling techniques for desktop two-handed input. In: Proc. UIST '98, ACM Press, pp. 49–58.

Hinckley, K., Cutrell, E., Bathiche, S., Muss, T., 2002. Quantitative analysis of scrolling techniques. In: Proc. CHI 2002, ACM Press, pp. 65–72.

ISO, 2000. Ergonomic requirements for office work with visual display terminals (VDTs) – part 9: Requirements for non-keyboard input devices (ISO 9241-9:2000). ISO.

Kabbash, P., Buxton, W., Sellen, A., 1994. Two-handed input in a compound task. In: Proc. CHI '94, ACM Press, pp. 417–423.

Krueger, M.W., 1983. Artificial Reality. Addison-Wesley, pp. 142–147.

Krueger, M.W., Gionfriddo, T., Hinrichsen, K., 1985. VIDEOPLACE – an artificial reality. In: Proc. CHI '85, ACM Press, pp. 35–40.

Latulipe, C., Kaplan, C.S., Clarke, C.L.A., 2005. Simultaneous rotation and translation. In: Proc. HCI2005, British Computer Society, vol. 2, pp. 63–67.

Latulipe, C., Kaplan, C.S., Clarke, C.L.A., 2005. Bimanual and unimanual image alignment: an evaluation of mouse-based techniques. In: Proc. UIST 2005, ACM Press, pp. 123–131.

Latulipe, C., Mann, S., Clarke, C.L.A., Kaplan, C.S., 2006. symSpline: bimanual symmetric spline manipulation. In: Proc. CHI2006, ACM Press, pp. 349–358.

Leganchuk, A., Zhai, S., Buxton, W., 1998. Manual and cognitive benefits of two-handed input: an experimental study. ACM Transactions on Computer-Human Interaction 5, 326–359.

MacKenzie, I.S., 1992. Fitts' law as a research and design tool in human-computer interaction. Human-Computer Interaction 7, 91–139.

MacKenzie, I.S., Buxton, W., 1992. Extending Fitts' Law to two-dimensional tasks. In: Proc. CHI '92, ACM Press, pp. 219–226.

Martin, B., Raisamo, W., 2004. TrackMouse: a new solution for 2+2D interactions. In: Proceedings of NordiCHI 2004: Third Nordic Conference on Human-Computer Interaction, ACM Press, pp. 89–92.

Nigay, L., Coutaz, J., 1993. A design space for multimodal systems: concurrent processing and data fusion. In: Proc. INTERCHI '93, ACM Press, pp. 172–178.

Schmidt, R.A., Zelaznik, H., Hawkins, B., Frank, J.S., Quinn Jr., J.T., 1979. Motor-output variability: a theory for the accuracy of rapid motor acts. Psychological Review 86, 415–451.

Soukoreff, R.W., MacKenzie, I.S., 2004. Towards a Standard for Pointing Device Evaluation, perspectives on 27 years of Fitts' law research in HCI. International Journal of Human-Computer Studies 61, 751–789.

Sutherland, I.E., 1963. SketchPad: A man-machine graphical communication system. AFIPS Conference Proceedings 23, 323–328.

Todor, J.I., Doane, T., 1978. Handedness and hemispheric asymmetry in the control of movements. Journal of Motor Behavior 10, 295–300.

Westergaard, M., 2002. Supporting Multiple Pointing Devices in Microsoft Windows. Microsoft Summer Workshop for Faculty and PhDs. Cambridge, England. <http://klafbang.dk/personlig/publications.php3/>.

Zhai, S., 2004. Characterizing computer input with Fitts' law parameters – The information and non-information aspects of pointing. International Journal of Human-Computer Studies 61, 791–809.

Zhai, S., Smith, B.A., T. Selker, 1997. Improving browsing performance: a study of four input devices for scrolling and pointing tasks. In: Proc. INTERACT '97, IFIP, pp. 286–292.